DOCUMENT RESUME

ED 279 305 IR 012 545

TITLE	The National LOGO Exchange. The LOGO Newsletter for Teachers. Volume 4 (September 1985-May 1986).
INSTITUTION	National LOGO Exchange, Charlottesville, VA.
PUB DATE	86
NOTE	159p.; For volumes 1-3, see ED 264 842.
AVAILABLE FROM	The National LOGO Exchange, PO Box 5341, Charlottesville, VA 22905 ($25.00 for one-year subscription).
PUB TYPE	Collected Works - Serials (022) -- Guides - Classroom Use - Guides (For Teachers) (052) -- Reports - Descriptive (141)
JOURNAL CIT	National Logo Exchange v4 n1-9 Sep-May 1985-86
EDRS PRICE	MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS	Book Reviews; *Computer Graphics; Elementary Secondary Education; *Learning Activities; *Merchandise Information; Newsletters; Program Descriptions; *Programing; *Programing Languages; *Teaching Methods
IDENTIFIERS	*LOGO Programing Language

ABSTRACT

This document consists of nine issues of the National LOGO Exchange Newsletter (NLX). Each issue contains several articles as well as regular features and departments. The lead articles in this volume are entitled: (1) "Twenty Powerful Ideas" (Molly Watt); (2) "Multiple Turtles without Sprites" (James Hurley); (3) "Teachable Logo Moments: Jimmy Comes to Class" (Louise Robinson); (4) "Come to the LogoShow!" (Doris Schroeder); (5) "Sometimes Logo Teachers Should Teach" (Marge Stouffer); (6) "Logo Teaching: A Marriage of Theory and Practice" (Ceil Humphreys); (7) "Euler and the Turtle" (Judi Harris); (8) "Procedure Inputs: Local Variables or Not?" (Sharon Burrowes); and (9) "Logo Quilt: A Collaborative Learning Project" (Dan Watt and Molly Watt). Additional articles and information on Logo are presented under such headings as: NLXtra (a supplement to NLX); NLXionary (selected readings, Griff Wigley); NLXual Challenges (Robs Muir); From the Editor (Tom Lough); Logo Teacher Feature (Rebecca Poplin); Logo Disserts (dissertations dealing with Logo, Barbara Elias); Listful Thinking (Glen Bull and Paula Cochran); Teacher to Teacher (software reviews, Anne Cairns Federlein, James Fry, or Regina M. Bowden); Tipps for Teachers (Steve Tipps); Turtle Tips (Jane Toth and Donna Lanyi); and Q and A (Jim McCauley). Information about conferences, workshops or courses on Logo is also provided, as well as announcements of educational computing broadcasts and information for ordering Logo related publications, curriculum materials, and resource packets. (DJR)

****
*   Reproductions supplied by EDRS are the best that can be made   *
*   from the original document.   *
****

ED279305

THE NATIONAL
LOGO
EXCHANGE
The Logo Newsletter for Teachers

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☒ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY
Tom Lough
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Twenty Powerful Ideas

by

Molly Watt

When teachers know of the powerful ideas embedded in Logo, they can teach more effectively. Powerful ideas influence how they introduce Logo, how they decide which examples to show, and how they establish the context and culture for their students. When teachers know and demonstrate more than just a knowledge of the Logo language and its syntax, students are able to use Logo more powerfully and intentionally.

I hope that this set of twenty ideas will give you some things to consider as you begin a new school year.

1. **The Turtle.** Everyone knows how to move their own bodies. By using that knowledge and identifying with the turtle, everyone can tell the turtle how to move across the graphics screen. By teaching the turtle, we learn how to program in Logo.

2. **Playing Turtle and Playing Computer.** Pretend to be the turtle and walk the design you want the turtle to draw on the screen. Observe carefully your steps, your directions, your turns. Translate your steps and turns into Logo commands. This is a good programming tool and an excellent debugging tool.

3. **The Total Turtle Trip Theorem.** Whenever the turtle has drawn a figure and returned to its starting location and heading, it has turned a total of 360 degrees (or some multiple) during the Total Turtle Trip. This idea contains information needed to draw any regular shape, such as a square (4 times 90) or a triangle (3 times 120).

4. **Drawing.** Almost everyone uses drawing as a natural way to express ideas. With Logo, you are using the turtle to draw and to express your ideas.

5. **Exploration and Discovery.** Every Logo user is an explorer and can make genuine discoveries of how to draw a shape or solve a problem in its rich mathematical environment. A successful explorer must be willing to try ideas and see what happens.

6. **Theory Building and Problem Solving.** Logo learners make up theories about everything, especially about turtle moves. Taking note of their specific expectations and of the differences between them and what actually happened invites students to develop workable problem solving strategies.

7. **Describing and Defining.** All Logo procedures are descriptions of something you want Logo to do. You can save these descriptions as procedures and use them over and over. Also, describing and defining a problem is useful in debugging or solving that problem.

8. **Debugging.** Trapping and eliminating bugs is a great deal of what Logo programming is about. (Ed. note: See NLXual Challenges elsewhere in this issue.) Hardly anyone gets a Logo program to work the way they want the first time through. This is the problem solving "curriculum" built into Logo. Logo error messages are helpful in finding bugs.

9. **Procedures.** When doing routine jobs, most of us follow a series of small procedures. A procedure does one small task in life, as well as in Logo. Procedures are Logo's helpers. They can be combined to work together with a superprocedure as a boss. Procedure trees show how procedures work together and in what order they are called.

IR012545

BEST COPY AVAILABLE

Powerful Ideas continued

10. **Naming.** Everyone loves to name things. In Logo, you name anything you want to save or use again. Naming helps you remember what you did and allows you to use something again by calling its name. Also, naming is an important part of problem solving.

11. **Extensibility.** You can teach Logo how to do something by defining a procedure. This extends the language.

12. **Variables.** A variable is a piece of information. It could be thought of as a place to put a number (or other object) as input to a command or operation. Numerical variables allow you to create the same shapes in many different sizes on the screen, thus giving you a great deal of power in using small building blocks.

13. **Repeating.** Logo's REPEAT command allows you to repeat a procedure or any list of instructions a number of times. With REPEAT, you can explore patterns, create tesselations, and discover mathematical relationships between numbers or variables.

14. **Recursion.** A Logo procedure can call another procedure by the same name (a copy of itself) as a way to get a job done simply. The idea of recursion is one of the most powerful ideas embedded in Logo, and it is not easy to grasp. Using recursive procedures to describe actions in real life is helpful to students. For example, let's ask Dan to clear the dinner table and wash the dishes. Using recursive thinking, Dan would check to see if there were any dirty dishes on the table. If not, his job is done. If there were, he would clear his own place and wash his own dishes, then call a helper with the same set of instructions. With such examples, students can grow into the more complex versions of recursion and come to use Logo's full power as an artificial intelligence language.

15. **Conditionals.** Logo can stop procedures when certain conditions are met. For example, IF HEADING = O [STOP] might be used to stop a recursive procedure. Of course, conditionals have many other uses as well.

16. **Interaction.** Logo as a process is interactive, and invites students to learn at the computer by trying things out. More advanced programming allows users to input information into programs with commands such as READCHAR, REQUEST, and READLIST.

17. **Information Processing.** Logo stores information as words and lists. Naming files and procedures and using REPEAT and PRINT are simple examples of processing words and lists. Logo can also take information input as words and lists, operate on it, and give the result as output.

18. **Thinking About Learning and Learning About Thinking.** Part of learning Logo is observing your own thought processes made visible on the screen, often in the form of turtle tracks. By watching the turtle's heading, you can see your own thought process in estimating angles. By describing it in words, you can refine your thinking about angles.

19. **Microworlds.** Logo makes it possible to establish a safe, defined learning environment where students can do some real learning through open ended exploration. Teachers become guides and challengers, rather than lecturers.

20. **Modes of the Computer.** Logo has two main modes: direct (or command) mode, and edit mode. When you are working in the direct mode and you type an instruction and press the "do it" key (sometimes called <return> or <enter>), something happens. In the edit mode, however, instructions are stored to be used later. Nothing happens when "do it" is pressed. Understanding the effect of each mode and being able to shift from one mode to the other is essential to learning to use Logo effectively.

(Ed. note: This set of ideas is the product of Molly and Dan Watt's thinking and current work in Logo. More detailed versions of this article appear in Molly's Welcome to Logo! series from D. C. Heath, and Molly and Dan's Teaching with Logo from Addison-Wesley.)

Molly Watt is a teacher, lecturer, and author in the field of educational computing.

Imagewriter Overprinting?

Are you frustrated by the Apple Imagewriter when trying to print listings of Logo procedures which have long lines? Glen Bull is actively seeking a way to prevent the overprinting which results. So far, no solution has been discovered. If you have ideas or suggestions, send them to Glen in care of the NLX. Meanwhile, here is a "clunky" fix.

1. Load in the file to be printed.
2. Bring the procedures into the editor. (Use EDIT ALL with MIT Logo or EDIT [SQUARE TRI MOVE ....etc.] with LCSI.
3. Locate any instruction lines which take up more than a single line in the editor.
4. Move the cursor to the middle of the second editor line of the long instruction line, type ! and press <return>. This has the effect of "breaking" the line, as far as the printer is concerned.
5. Repeat step 4 as often as necessary, until all long lines are "broken."
6. Define the procedures by leaving the editor appropriately.
7. Print out the file. The Imagewriter will print the "broken" lines with an exclamation point at the end of each to indicate that they are actually continued.
8. Erase the file. It will not run properly with the "broken" lines. If you want to run procedures in the file, load them from the disk once more.

NLXtra
A Supplement to the National Logo Exchange Newsletter

IF: At First You Don't Succeed

by
Michael Tempel and Mario Bourgoin

This is the last of four NLX articles about writing your own version of IF in Logo. After presenting the problem in the March issue, we gave a few solutions in April and extended the discussion in May.

The response to the articles has been strong. Many points of view came through, each expressing a different style of thought. All the user created IFs depended on the fact that Logo predicates output the literal words TRUE or FALSE and that these words may be the names of things in Logo.

MICHAEL'S.IF, shown in the April NLX, used the words TRUE and FALSE as LABELs for GO. (Similarly structured IFs were sent in by Sharon Burrowes and Eric Nagler.)

BRIAN'S.IFF and YVES'.IF use TRUE and FALSE as the names of lists which are then RUN. This technique was also used by Jan Bybee, Forrest Smith, Lorana Jinherson, and Patrick Salsich.

Sylvan Wallach submitted an IF that relies on the use of TRUE and FALSE as the names of procedures:

```
TO IF.THEN.ELSE :COND :IF.ACT :ELSE.ACT
RUN (SENTENCE :COND)
END

TO TRUE
RUN :IF.ACT
END

TO FALSE
RUN :ELSE.ACT
END
```

Solutions using Sylvan's general approach were also sent in by Robs Muir and Tom Lough.

Jan Bybee and Forrest Smith came up with an interesting variant:

```
TO IFELSE :CONDITION :TLIST :FLIST
DEFINE "TRUE [ [ ] [RUN :TLIST] ]
DEFINE "FALSE [ [ ] [RUN :FLIST] ]
RUN (LIST :CONDITION)
ER [TRUE FALSE]
END
```

Instead of writing the procedures TRUE and FALSE themselves, they have IFELSE create them using the primitive procedure DEFINE. One of these two procedures is run, and then both are erased.

IF OPERATIONP

In our April article, we raised the question of whether or not a user defined IF could be either a command or an operation like the primitive IF. Here's a solution written in Apple Logo II.

```
TO IF :COND :TRUE :FALSE
CATCH "ERROR [OUTPUT RUN THING :COND]
END

?IF 3=3 [PRINT "HI] [PRINT "BYE]
HI
?PRINT IF 3=3 ["HI] ["BYE]
HI
```

What's going on here? Well, the key is RUN THING :COND, which is the same as BRIAN'S.IF. Now, if we treat IF as a command, as in the first example, we get the desired result, but RUN (which may be a command or an operation) doesn't report (or "output") to OUTPUT. This error is caught by CATCH "ERROR so no complaint is printed and program execution continues.

If we treat IF as an operation, as in the second case, there is no error, so CATCH "ERROR has no effect. Unfortunately, all errors are caught.

IF LEARNINGP

Using a strategy similar to ours, Cynthia Solomon also wrote an IF that behaves either as a command or operation. But, in her letter, she raised an important issue: "In the case of developing a procedure for addition when there is no ADD primitive[1], I see an opportunity for learning a lot about computing and about addition, even if I end up with a limited ADD procedure. In this case, ... I am left wondering what you expect people to learn from the experience. What are the important ideas that this problem illustrates? Personally, I became reacquainted with the differences [between] Atari and Apple Logos."

In fact, there was no initial goal. Some of us simply became interested in the problem and pursued it. Partly, it was just the pleasure of the challenge ... the doing of something that first appeared impossible. Since we found it engaging, we figured other Logo people might also, so we presented it in these articles.

In hindsight, what has been the value of this exercise? What have we learned about Logo? What have we learned about other domains?

Along with Cynthia, one thing we have learned is that there are differences between Logos. These may appear minor at first, but they are really philosophical at the core. As with any high level language, Logo wants you to concentrate on the problem at hand, not on how Logo is implemented. Because you can create an IF, you should expect Logo's IF to behave as yours will. Those of you who have Apple Logo or an MIT Logo know that this is not the case[2]. Should Logo support more English-like features? Part of Logo's power is that it asks you to say exactly what you mean. How can Logo be more intuitive and yet remain unambiguous?

Most versions of Logo support an IF that takes an optional input. This is possible because those who write Logo have access to features that are not available without machine language level tinkering. Optional inputs can make Logo programs hard to understand and make debugging difficult. You might be better off using Lisp. Its parentheses make clear which procedures get what inputs and from whom. This is called "parsing."

A more technical issue is the difficulty for Logo users to create new utilities that work correctly in all cases. Remember the problem of dealing with errors in user created versions of IF? This points out large holes in Logos. Many primitives are missing, such as those that allow more complete error handling or the generation of your own error messages. Try writing a STEP procedure. Does it handle errors properly?

On the positive side, we learned about the power of naming in Logo. All of the solutions rely on the fact that the words TRUE and FALSE may be the names of lists, procedures, property lists, and labels. Logo allows you to name something to be done in all of these ways. This is because there is little difference in Logo between data and procedures. A list can be seen as a container of words or as something to do.

We could now produce an IF that includes a MAYBE case and move into the world of multi-valued logic. Such an IF would still work with Logo's predicates that return TRUE and FALSE, but we would add new predicates that return this name MAYBE in special circumstances.

IF DECISIONP

We have also found out something about how Logo makes decisions. The trick is deciding what to do in the TRUE case, and what to do in the FALSE case. A predicate is used to find out which case applies and we then RUN the action that this case names. (How much does this relate to how people make decisions?)

This idea can be used in other circumstances. For example, Yves Poiree wrote a straightforward way to change the penstate.

```
NAME [PENDOWN] "PENUP
NAME [PENUP] "PENDOWN

TO CHANGE.PEN
RUN THING FIRST PEN
END
```

What's happening here? We name what to do if the pen is up and what to do if it is down. FIRST PEN tells us which is the case so it acts as the predicate. Then we find the thing named by the penstate and RUN it.

We hope you have enjoyed the IF articles. Please share your further explorations and discoveries with us. Who knows? Perhaps you will be the first one to write an OUTPUT procedure in Logo!!

1. See: Solomon, Cynthia J. Teaching the Computer to Add: An Example of Problem Solving in an Anthropomorphic Computer Culture. Logo Memo No. 41, 1976, MIT AI Lab.

2. See: Harvey, Brian. Computer Science Logo Style. MIT Press, 1985. Appendix A contains a thorough discussion of the differences between the various versions of Logo.

Michael Tempel is Director of Training with Logo Computer Systems, Inc., 555 West 57th Street, Suite 1236, New York, NY 10019. (212) 765-4780. Mario Bourgoin is a Research Assistant with the Learning Technologies Group, Arts and Media Technologies, Massachusetts Institute of Technology, Building E15-309, 20 Ames Street, Cambridge, MA 02139. (617) 253-8579.

From The Editor

by Tom Lough

Hello and welcome back to school! I hope you are excited about the coming months and the adventures which await you and your students! I look forward to sharing them with you.

The summer was a rewarding one. Our class in Hawaii concentrated on developing Logo tools for teachers in language arts, mathematics, and science in the middle grades. After seeing some of the tool kits, I want to be a 6th grader all over again!

The Logo 85 conference at MIT was as exciting as ever. I was pleased to note an increased international interest in Logo.

At the World Conference on Computers in Education, we co-sponsored a "Logo Day," with four panels and two informal meetings. These activities were characterized by healthy attendance and enthusiastic interaction. Steve, Glen, and I also assisted Joyce Tobias and Tim Riordon in a series of well received Logo workshops.

Finally, I was privileged to participate in one of LCSI's Workshops for Experienced Logo Users, in which important teaching issues were discussed and advanced topics explored. I caught glimpses of where Logo is going in the next few years. Oh, my! BASIC was never like this!

Throughout all these activities, I was impressed with the zest and continuing enthusiasm of the "Logo movement." I saw content area teachers struggle with new Logo concepts, then light up as the ideas came for using them in their classes.

Logo is moving into the high schools and into the content areas for a variety of reasons. Perhaps the most influential reason is because there is a growing group of students who have had Logo when they were in elementary school, and are now demanding it as they progress through the higher grades. And that's OK. Children have led adults through many important movements. Why not an educational one?

It seems as if each year becomes better than the one before. If this is what Logo does, then let's get started!

FD 100! Tom

Tipps for Teachers

by Steve Tipps

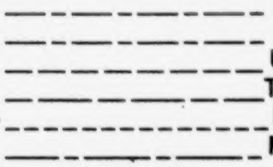
Take a few minutes for your children to look at the art of Van Gogh. The bold slashes of colors result in some of the most striking and beautiful art. George Seurat, the pointillist painter, used thousands of dots to create giant and beautiful pictures. Here are two examples of big things made of many small parts.

Children can find many other examples of patterns made of small parts. You can have them look at dot matrix print from a computer or dots used to make time-and-temperature signs. Investigate the composition of newsprint and comic books with a magnifying glass.

These examples of dots and dashes may lead to Logo projects which offer challenges to Logo learners from primary through adult. For the new school year, here are some projects which might be used to introduce or review important concepts and skills in Logo.

DASH AWAY ALL

Many children may have created a dashed line effect using PENUP and PENDOWN with FORWARD. By varying the length of the PENUP and PENDOWN parts, the lines takes many forms.



DASHED LINES

USE PENUP AND PENDOWN TO DRAW SOME OF THESE DASHED LINES. THEN TRY MAKING YOUR OWN LINES.

A few examples like these should inspire a hundred others. If children already know REPEAT, they see how useful it is in this case. If they do not know REPEAT, making dashed lines is an excellent way to create that need.

The patterns which are drawn can be compared.

```
REPEAT 10 [ FD 15 PENUP FD 10 PENDOWN]
REPEAT 5 [ FD 20 PENUP FD 5 PENDOWN FD 10
PENUP FD 5 PENDOWN]
REPEAT 8 [ FD 10 PENUP FD 50 PENDOWN FD 20
PENUP FD 10 PENDOWN]
```

Tipps continued

Investigate which part of the pattern is drawn and which part is "skipped." Some children may have a hard time making this distinction. You can help by posing questions about the line and the skips. Which part makes the dash? Which part makes the space?

The social interaction, discussion, and argument between children in matching lines with REPEAT patterns are excellent means for learning that the turtle has two different functions. Seeing the turtle as a drawing tool and as a position mover is important as children gain mastery in Logo. It is quite possible that your children will define the space as the DASH, because the space interrupts the line and makes it dashed. Eventually this may get in the way of DASH projects, but let them work through that assumption.

Another observation may be that the line made by certain combinations of PENUP (PU), PENDOWN (PD), and FORWARD (FD) are not "accurate." For instance, PU FD 10 PD FD 10 makes a line which may not be even. The dash made by FD 10 may be longer than the space make by FD 10. This is a result of what I call "pixel capture." The lighted pixels of picture elements on the monitor sometimes capture nearby pixels and turn them on as well. While this is not a very technical explanation, most children will be familiar with the idea that a light bulb in a dark room also lights up the area around it.

DASHING COMMANDS

Rather than making a procedure of an entire line or repeat statement, use a single movement as the basis of a procedure. Writing a DASH procedure is an easy but important task, when familiar with PU and PD. After working with the dashed line, children may decide that a dash is about 5 turtle steps long. They can capture this idea in a DASH procedure. In fact, DASH may be a very good procedure to introduce the concept of creating your own commands. It is simple, functional, and modular. By making DASH the drawing part of a pattern, many lines are possible.

```
TO DASH
PENDOWN FD 5 PENUP
END


REPEAT 10 [ DASH FD 20]
REPEAT 10 [ DASH DASH DASH FD 10]
REPEAT 10 [ DASH FD 10 DASH DASH FD 10]
```

A length of 5 may be too big for some tastes, so that 2 or 3 is preferred. Different dashes may be better for different projects. The variability of the dash size is something which children can see. A variable DASH may be just what the turtle ordered. If children invent the need for variability, the variable form is amazingly simple for them to grasp, and incredibly versatile as well.

```
TO DASH2 :LENGTH
PENOWN FD :LENGTH PENUP
END
```

Now REPEAT statements can be written with the plain DASH or a variable DASH.

```
        REPEAT 10 [ DASH DASH FD 10 DASH FD 20 ]
or variably
        REPEAT 10 [ DASH2 10 FD 10 DASH2 5 FD 20]
```

Some projects don't work quite as well if the displacement of the turtle is 2 or 3 turtle steps. A third DASH procedure returns the turtle to its original position.

```
TO DASH3 :LENGTH
PENDOWN FD :LENGTH BK :LENGTH PENUP
END
```

The projects which children invent determine what type of DASH is most useful. I found that DASH, DASH2, and DASH3 were equally useful to me and allowed me to do different things. Multiple DASH procedures extend the understanding of procedural thinkng. Executing several commands by creating and naming a procedure is powerful; using this power is one key to unlocking Logo.

DASHING SHAPES

Dashes can be used with familiar shapes. In fact, children who already "know" square, triangle, and circle may have to rethink their ideas if they use DASH to complete the figures.

Think about the differences between

```
        REPEAT 4 [ DASH FORWARD 50 RIGHT 90]
and
        REPEAT 4 [ FORWARD 50 RIGHT 90 DASH2 5]
and
        REPEAT 4 [ FORWARD 50 DASH3 5 RIGHT 90]
```

Part of the answer lies with the kind of DASH which is defined. But part also lies with the order of

Tipps continued

execution of the REPEAT command. The objective here is not drawing a square, but thinking about squareness. Try the same with a triangle, hexagon, elevenagon, and circle shapes. Your results may surprise you.

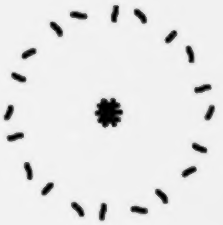
SPINNING DASHING SHAPES

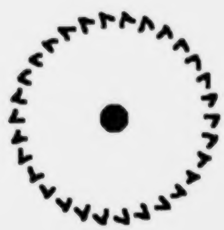
Now try rotating or spinning the shapes. For example, create your best dashed triangle. Then rotate it using a REPEAT statement.

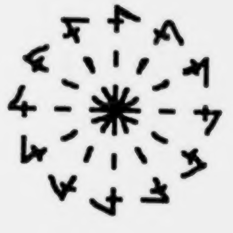
```
TO DASH.TRIANGLE
REPEAT 3 [ DASH FD 40 RIGHT 120]
END
```

```
REPEAT 10 [ DASH.TRIANGLE RT 36]
REPEAT 24 [ DASH.TRIANGLE RT 15]
REPEAT 30 [ DASH.TRIANGLE LT 12]
```

Use different dashes to see what the results are and create several different DASH.TRIANGLE descriptions. You may find some surprising results.







Try designs with DASH by spinning the square, then by spinning the square and the triangle together. Spinning a shape may be trial-and-error at first. With experience and a little guidance, students can learn to stop the turtle exactly where it starts. With sufficient experience, the importance of 360 may be found and used for planning.

GETTING THE POINT WITH DASH

Explorations with DASH offer many possibilities for learning with Logo. Dashes are modular in nature. One DASH by itself may not be very interesting. But when combined with other moves, wonderful things can happen. Dashes allow us to look at parts of a whole. Seeing old things with new eyes is very exciting.

Another possible learning would be to reduce the size of DASH to FD 0.1 or FD 0.0001. Children might agree that the turtle had moved, but that you could not see the movement because it was so small. This realization provides an opportunity to introduce the concept of point--a position in space that has no dimension. Even though mathematics textbooks give that definition, a point is always represented as a period on the page. Logo is a vehicle for reducing the size of a DASH to naught. In fact, it might be a wonderful occasion to find out how small a number your version of Logo takes without complaint. With Apple Logo II, I could type

```
DASH2  0.000000000000000000000000000001
```

and still get a dot on the screen. The dot, or pixel, is the smallest unit the computer can use to represent a point.

At the Logo 85 conference, Seymour Papert said that the real test of Logo may be whether it allows children to think in their own way rather than being confined to thinking in some prescribed manner about problems. DASH is adaptable to the skill and needs of children at many levels and gives them both power and freedom to think.

DASHING OFF FOR THE NEW SCHOOL YEAR

As you work this year, I would really like to hear from you. I learn so much from the questions, projects, and frustrations which you share with me. They keep me thinking about new ways with Logo. I would especially like to hear from your students when you have tried out one of the ideas in the column. Just put out an envelope in your classroom for student projects and send them to Steve Tipps, Midwestern State University, Wichita Falls, Texas 76308. Tell me how your class uses the ideas you share. I'd love to see them. Thanks!

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX. He conducts Logo workshops for school systems throughout the United States,

Logo Teacher Feature
by
Rebecca Poplin

Featuring: Robs Muir

Challenges are a vital part of Robs Muir's life. NLX readers are familiar with some of the Logo challenges which Robs incorporates into his instructional environment. These NLXual Challenges are a natural outgrowth of one realm of activities in which Robs is involved. Robs and his wife Candi teach fifth and sixth graders in a small elementary school in Claremont, California. Their students are actively involved in solving structured problems with many possible solutions.

A critical phase of the problem-solving is sharing the solutions with others. Robs has placed increasing emphasis on the sharing phase of problem-solving as he has worked more with Logo and has realized the benefits of this process.

Two of the most exciting problems last year produced a Logo program which created a word search puzzle and several versions of "Stationmaster," a favorite indoor game. (Ed. note: See NLXual Challenges in the February 1985 NLX.) Problems and solutions are shared within the classroom and with NLX readers all over the United States and in other countries.

Robs has recently taken on a new challenge. He has become the coordinator of a project supported by the Apple Education Foundation, in which the Logo Conference Tree (a public telecommunications system for sharing information and ideas about Logo) was established. Ten school districts in southern California are linked through this project. Students may use the service from their schools or homes. NLX readers are welcome to contact the Logo Conference Tree (300 baud) at (714) 621-2310. No identification or password is needed. Most of the NLXual Challenges are available through this service.

Robs also teaches at the Claremont Graduate School, where Logo is used extensively in teacher education and for teaching data structures in graduate computer science programs. He is also participating in planning the West Coast Logo Conference, scheduled for November 21-23, 1985.

In the coming school year, Robs is looking forward to the new challenges he has set for himself and his students. He expects to involve students more extensively in telecommunications. A persistent problem has been how to fit Logo into a time schedule in which certain allocations are mandated for various curriculum areas. As a result, Robs is interested in greater integration of Logo and curriculum areas, particularly science.

Teaching with Logo is a growing and learning process. Robs Muir conveys a sense of developing perspective, as he has modified his approach and use of Logo over time. In the past, Robs felt that every student should program in Logo, and his approach to computer instruction and use was more prescriptive than it is today. Challenge problems are shared, and many students participate in solving and creating them, but the computer is used in a variety of ways in addition to Logo.

Effective Logo teachers are constantly learning, thinking, and growing. They probe the environment for ways to stimulate creativity and growth with all the tools available, including the Logo language. Each month, this column will profile a teacher who is using Logo in his or her classroom on a regular basis. Readers are invited to suggest names of teachers who have developed an interesting or unique perspective on Logo. Please contact: Rebecca Poplin, 2421 Fain Street, Wichita Falls, Texas, 76308, or call (817) 766-5323.

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

NLXionary
A Lectionary of Selected Logo Readings
with Commentary and Opinion
by Griff Wigley

"Disorganized Students Do Better with Computers, Says Logo Creator." Conference Report, Education Computer News, June 19, 1985.

Seymour Papert sparked more controversy this summer at a Tufts University symposium on using computers in school science labs. The symposium was sponsored by the Technical Education Research Center (TERC) based in Cambridge, MA.

Papert reportedly argued that a systematic step-by-step approach to programming is not best for all students; yes, even that some of the best programmers are non-planners who fiddle with a program until it works. Papert believes that "messing about" is particularly beneficial in science classes because students learn more through exploration than through being forced to prove theories. He added that writing without using an outline may not be always bad either.

There you go again, Seymour. Taking pot shots at the established methods of teaching that have a solid history of alienating generations of students. Haven't you been keeping up on all the latest in problem solving? What about all those thinking skill hierarchies,

NLXionary continued

matricies, and taxonomies that educators everywhere are transforming into handy-dandy scopes and sequences? If you insist, "messing about" can be covered under creative thinking, which is taught only after the problem solving strategy of working backwards is mastered (usually by the 4th grade in progressive districts).

How can a teacher allow a student to "mess about" with a science concept for days, weeks, or as long as it takes to "sink into the subsoil of the mind?" What about chapters? Units? Tests? Quarters / Semesters / Trimesters?

Seymour, you sound as if you've been reading "In Search of Excellence," as if Managment By Wandering Around (MBWA) should become Learning By Wandering Around (LBWA). Most schools / corporations want their students / employees focused on the grades / profits for this quarter so the parents / shareholders will be happy. "Messing about" / "Wandering around " might make it in the free enterprise system where a company has incentives and the customer has some power. But never in the monopoly of public education where incentives are few and student / parent customers have relatively little power. Unless...

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Logo Disserts:
Dissertations Dealing with Logo
by Barbara Elias

Portuguese schools are now beginning to use microcomputers. Some preparatory and secondary schools now provide extra-curricular programming courses on a limited basis and a few pilot projects have been started. Recently, the Portuguese Ministry of Education formed a committee to work on ways to introduce computers into the schools. One important step in assuring an effective program is to prepare teachers to use this technology. A study has been conducted which presents the design of a Logo unit for prospective elementary teachers. It is offered as one means of helping to prepare teachers to teach Logo.

The study includes a literature review of Logo research and teacher computer education. Also included is a Logo course and an implementation and evaluation plan.

The literature summarized in this study suggests that (a) instruction in a programming language is an important part of course work which prepares teachers to deal with computer education, (b) Logo can provide valuable learning experiences for children, but the claims regarding logical thinking and problem solving skills remain inconclusive, and (c) Logo research in the area of teacher preparation is sparse and inconclusive.

The unit design includes a description of the learners who will use this unit, the content of the course syllabus, and resources and conditions desirable for implementation.

Learners expected to use this unit are first year college students, 18 to 21 years old. Most of them are female, and come from a variety of backgrounds. It is assumed that they are capable students who recognize the importance of computer education. The proposed program goals are the development of (1) Logo programming mastery, (2) positive attitudes towards mathematics and the teaching of mathematics, and (3) problem solving skills.

The proposed syllabus contains a wealth of information. Section I includes an introduction to Logo, turtle graphics, and workspace and file management. Section II presents advanced programming concepts, such as recursion, and a brief introduction to list processing.

Learning activities and teaching strategies are included in lesson plans organized around the program goals. Though fairly structured, they are intended for use with freedom and flexibility in a manner which helps to create a Logo learning environment.

Formative and summative evaluation measures are included. The researcher commends this study for consideration by Portuguese officials as one possible way of carrying out two important tasks now needed: curriculum development and research.

>>>Moreira, Candida Maria De Almeida Pauperio Queiroz (1984). Logo: Design of a Unit for a Course in Computers in Education for Portuguese Student Teachers of Escolas Superiores De Educacao. (Unpublished thesis, Master of Education, Boston University, 1984.)

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of Virginia.

Listful Thinking

by Glen Bull and Paula Cochran

Blending Logo and Language Arts

Welcome back! For new readers who are joining us, there are two tenets which underlie this column:

(1) We are more interested in using Logo to teach than in teaching Logo. Therefore the focus of the column is on content areas such as English rather than on Logo programming.

(2) We think Logo tools are essential for allowing the teacher to focus on content rather than programming. A Logo tool is a procedure which can be used by the teacher and by students as if it were built into the language. (Logo is one of the few programming languages which allows this to be done easily.)

At the beginning of school, an essay on "What I Did Last Summer" is almost obligatory. This summer we spent some time working with teachers at Punahou School on the island of Oahu in Hawaii.

We had fun exploring Oahu by car. Quickly we noticed that the street names all seemed to begin with K or H, and were about 40 syllables long with names like "Kanehamenamehahahakelike".

We tried different strategies. The navigator would tell the driver, "In about two blocks, turn to the right at 'Kao .. oo??' ." The driver would respond, "What?". Then the spelling strategy was invoked: "The first four letters are 'K A N E' and the last letter is also an E". We began to wonder, "Why are all the names so long?"

THE TOOL PHILOSOPHY

It turns out that the Hawaiian language has an interesting property. It has fewer sounds than most other languages. Since there are fewer sounds in Hawaiian, fewer combinations are possible. To create new words using the same sounds, the words must be longer. We wondered if we could write a Logo procedure that generates Hawaiian-like words. This led eventually to some explorations of English.

There are only six or seven consonants in Hawaiian. Here are the ones we could remember.

```
TO CONSONANTS
OUTPUT [P L K M N H W]
END
```

Now when we wanted to see what the Hawaiian consonants were, we simply typed:

```
?PRINT CONSONANTS
 PLKMNHW
```

(Throughout the column, the part you type is shown in boldface.)

The vowels in Hawaiian are the same as English.

```
TO VOWELS
OUTPUT [A E I O U]
END
```

We can look at the vowels in the same way.

```
?PRINT VOWELS
 A E I O U
```

We wanted to create a Logo procedure that would pick Hawaiian consonants and put them together with Hawaiian vowels to make Hawaiian syllables. Readers from last year will remember a common Logo tool that picks items from a list.

```
TO PICK :LIST
OP ITEM ( 1 + RANDOM COUNT :LIST ) :LIST
END
```

(In most versions of Logo, ITEM and COUNT are built-in. If you use version 1 of Terrapin Logo, you will need to use the procedures at the end of the column.)

Reflecting our tool philosophy, you don't have to understand what happens inside PICK. Try using it this way:

```
?PRINT PICK [X Y Z]
Y

?PRINT PICK [X Y Z]
Z
```

The procedure PICK chooses random items from a list, so you may have gotten different letters.

Now we can pick a Hawaiian consonant and a vowel.

```
?PRINT PICK CONSONANTS
M
?PRINT PICK VOWELS
U
```

Listful Thinking continued

```
?PRINT PICK CONSONANTS
H
?PRINT PICK VOWELS
I
```

If we put the letters that we picked together, it creates a Hawaiian word of sorts: MUHI.

This probably isn't a real Hawaiian word, but it looks and sounds like one. This word has a pattern. It consists of a consonant followed by a vowel followed by another consonant followed by another vowel. Linguists have a convention in which they use the letter C to stand for a consonant and the letter V to stand for any vowel. The pattern could be represented in this way.

<u>Consonant</u>	<u>Vowel</u>	<u>Consonant</u>	<u>Vowel</u>
C	V	C	V
M	U	H	I

We are going to represent a Hawaiian consonant with C and a vowel with V. We can do this with Logo procedures that work like this.

```
TO C
OUTPUT PICK CONSONANTS
END

TO V
OUTPUT PICK VOWELS
END
```

Now we can print a consonant or a vowel by typing:

```
?PRINT C
H
?PRINT V
A
?PRINT C
 M
?PRINT V
A
```

This makes another word that looks like it <u>might</u> be Hawaiian: HAMA.

Until now, we have had to put the separate letters together to see what the word looks like. There is a primitive procedure (WORD) built into Logo which will put the letters together for you.

```
?PRINT  ( WORD C V C V )
KIHU
```

Now that we have a debugged Hawaiian word generator, we can turn it loose!

```
?PRINT ( WORD C V C V C V C V C V  )
HAMAHANOHI
```

SPELLING LESSON

Thinking about Hawaiian and its permissable sound combinations made us start thinking about sound combinations in English. For example, English words, unlike Hawaiian, frequently begin with consonant blends or clusters.

Examples of blends which occur frequently in English (and therefore in spelling lessons) are BR, SL, ST, PL, etc. Can we adapt our Hawaiian word "generator" for English?

First , let's change the consonants to include sounds that often occur in English blends (especially R, L, and S). Edit your CONSONANTS procedure to look like this:

```
TO CONSONANTS
OUTPUT [ P L R S T B  D W]
END
```

To print consonant blends, try the following:

```
?PRINT (WORD C C V )
BTA

?PRINT (WORD C C V )
STO
```

AGH! Some of these combinations are not possible in English!

Even without formally studying the syllable structures and sound combinations of English, native speakers have knowledge about such things. You know which of the samples above are well-formed without reference to explicit rules. An exercise which emphasizes the consonant blends of English can take advantage of the intuitions your students also have about language. The goal is to make them more aware of their intuitive knowledge of English. This can make "rules" seem more obvious and easy to use.

Your students could test some randomly generated CCV combinations, using their own intuitions and a worksheet from you for guidance:

COMBINATION	ENGLISH BLEND?
___________	YES or NO
___________	YES or NO

Listful Thinking continued

They should type:

```
?PRINT ( WORD  C  C  V )
```

to get combinations to write on the worksheet. (Be sure to tell them about the control key for your version of Logo that retypes the line, so they only need to type the line the first time.)

Younger children may have trouble typing PRINT (WORD C C V) without errors. You can make a procedure for them if you wish.

```
TO BLEND
PRINT ( WORD C C V )
END
```

There is both an advantage and a disadvantage to creating a BLEND procedure. The advantage is that the teacher doesn't have to discuss the convention of using C and V to represent consonants and vowels. However, if students are given a BLEND procedure, they are less likely to experiment with other combinations such as:

```
?PRINT ( WORD C C C V )
or
?PRINT ( WORD C C V C )
```

Students can choose a couple of combinations that could be words in English. They can create definitions for the ones that sound interesting. For example, PRU: The sand left in the bottom of your sandals after a trip to the beach.

After several samples are generated, you might issue a challenge. Can your students change the letters in the CONSONANTS procedure so that no blends are produced? For example, we tried changing the CONSONANTS to:

```
TO CONSONANTS
OUTPUT [P T B N X Q]
END
```

This change caused combinations like this to be generated:

```
?PRINT ( WORD C C V )
XQA
?PRINT ( WORD C C V )
BNE
```

Can your class observe a rule about consonant blends in English? What sounds occur most frequently in blends? What is the maximum number of consecutive consonants which can begin a word? What will those consonants be? Try a P or a T in the blank spots: S _ L or S _ R.

We hope you will find that Logo blends beautifully with language arts and the exploration of language. Aloha!

Procedures for Version 1
of Terrapin Logo

```
TO COUNT  :LIST
OP COUNT.LOOP  :LIST 0
END

TO COUNT.LOOP  :LIST NUMBER
 IF  :LIST = [ ] OP  :NUMBER
 OP COUNT.LOOP BF  :LIST  :NUMBER + 1
END

TO  ITEM  :NUMBER  :LIST
OP  ITEM.LOOP  :NUMBER  :LIST  1
END

TO  ITEM.LOOP  :NUMBER  :LIST  :COUNT
 IF   :LIST = [ ] OP [ ]
 IF   :NUMBER  =  :COUNT OP FIRST  :LIST
 OP  ITEM.LOOP :NUMBER  BF :LIST :COUNT + 1
END
```

Glen Bull is a professor in the University of Virginia's School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a reformed English major who studied linguistics at the University of Cambridge. She is now a speech-language pathologist working with language-disabled children.

ILX Announced

Posy Publications announces the establishment of the International Logo Exchange (ILX) newsletter! Scheduled to begin publication in January 1986, the ILX is designed for those who want to keep abreast of Logo activities from an international perspective. Continental field editors will write columns for each ILX issue, highlighting recent Logo activity in their areas. Each issue will also contain notices of conferences, trips, and other activities of international scope. Editor of the bimonthly publication will be Dennis Harper, of the University of California at Santa Barbara.

A subscription to the ILX costs $19.50 per year for addresses in the USA, Canada, and Mexico, and $24.50 per year elsewhere. The International Logo Exchange, PO Box 5686, Charlottesville, VA 22905.

TurtleTips

by
Jane Toth and
Donna Lanyi

Breathe fire into the Logo activities in your classroom as the year begins, and get your students off to a "hot start" in programming! Here are more Logo activities, ideas, and procedures to help motivate your students.

The Turtle Teacher

Perhaps your students worked with Logo last year. But how much do they remember? As in all subject areas, student retention tends to slip over the summer months. We have found that the beginning of the school year is an excellent time to use pre-computer activities with students to help their recall and gain further mastery of concepts. We work many of these activities into other subject areas, too.

Often, computer lab instructors wish they had more time for these activities. Why not a partnership between classroom teacher and computer teacher to accomplish this? Design appropriate Logo activities, games, or tasks that can be done during math, reading, or other subjects. After all, teachers have always been expert at creating games to enhance a concept. Here are some ideas based on other popular games you probably have used before.

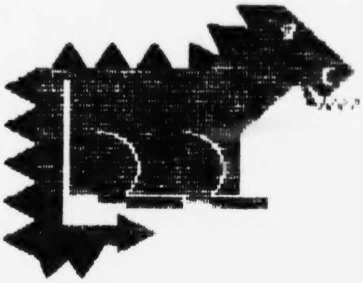
Turtle Hunt is a variation of Treasure Hunt. Have students work in small groups to design turtle maps with directions to secret locations given in Logo commands only. After the maps are completed, have groups trade maps and try to locate the hidden "turtle treasure" (prizes or treats).

Bingo, Concentration, and popular board games can be revised to include Logo commands. Turtle Bingo cards containing Logo commands could be used with turtle shaped markers, for example.

The Yuppies have us all intrigued wih the game of Trivia Pursuit. How about a Turtle Challenge board game along the same line? When presented with questions about Logo, students accumulate points or move turtle markers according to their answers, trying to "go home.

Procedure of the Month

Use the dragon's fire to ignite the imagination of your students! They enjoy designing and drawing all sorts of fantastic creatures with Logo, learning as they go. This dragon "starter" picture was submitted by Pat Arnold, a first grade teacher here in Wooster. If you are interested in a listing of this program, please send us a self addressed stamped envelope.



Last year, we were overwhelmed by your excitement over the "Procedures of the Month" and all the possible applications. We thank you for your responses, and invite you and your students to send us ideas and procedures for use in this column.

Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

Teacher to Teacher

A Review of New Logo Materials

by Anne Cairns Federlein

Fall is here and, with the new school year, Logo teachers discover they have a greater choice of publications to use as supplements to their Logo curricula. There are still the plethora of books addressing the needs of beginning students. However, some authors have developed texts that take advanced students and teachers into new and unexplored territories.

In order to assist teachers with all levels of students, this column will review a wide range of publications each month and define their strengths. These new and exciting materials offer a guide for facilitating learning through discovery in the classroom and the establishment and nurturing of exciting learning environments.

Teacher to Teacher continued

Beginning Students

The Workbook for Learning Logo, by Annie Kwok, Robert Sones, and Leon Hochglaube, is published by Turtle Publishing, a division of Art-I-Zan Specialities, Inc., 4810 Mariette Avenue, Montreal, Canada H4V 2G1.

Target Audience: elementary age children

Content: The workbook includes chapters on Getting Started, Meeting the Turtle, Screen Modes, Colors, and Procedures. It is designed to be used with a workbook diskette available from the publisher for $15 (which, by the way, includes several utility procedures in addition to Logo programs for the book).

Strengths: The text is written in an easy to understand style and encourages both early and later elementary children to make discoveries on their own. The illustrations and text size are very appealing to both adult and child, and offer a change from written instructions. The sequential format allows the teacher to add creative classroom activities at each discovery in Logo. There are also blank pages and wide margins on each page to encourage the children and teacher to take notes on their work in each chapter. A cardboard turtle and turtle board game are included to help introduce the concept of turns and degrees.

Advanced Students

Beyond Mindstorms: Teaching with IBM Logo, by Joyce Tobias, Jerry Short, Sharon Burrowes, and Tom Lough, published by CBS College Publishing, Holt, Rinehart and Winston, 383 Madison Avenue, New York, NY 10017.

Target Audience: Logo teachers

Content: The title indicates that this book goes beyond Seymour Papert's Mindstorms "not in terms of ideas, but in terms of applications. It presents an approach designed to help teachers learn to use Logo and then to teach it to their students and to use it to enhance the curricula." The three divisions of the text include:

(1) Welcome to Logo - an introduction and tour of IBM Logo,

(2) IBM Programming - discusses graphics, non-graphics, the editor, pictures and words, saving work, subprocedures, random events, recursion, inputs to procedures, and global variables, and

(3) Beyond Programming - addresses creating a Logo environment, the Logo curriculum, teaching wih Logo, working with special students, and the future of Logo.

Strengths: The value of this book is that it takes both teacher and student to a higher level of learning in Logo. Many of the publications on the market today only address the immediate technical and curricular concerns of the beginning student and teacher. These authors discuss the practical concerns of beginners and then proceed to integrating Logo into the school curriculum in different subject areas and grades, not just in the computer classes.

Once the practical aspects of Logo have been concretely presented, the authors take us to the philosophical level and outline the future of Logo. The Logo microworld offers important models for other educational environments and emphasizes the enjoyment of learning, the sharing of ideas, the importance of the gradual accumulation of fragments of knowledge, the discovery method of learning, and the acceptance of the outcomes during discovery learning.

Teachers could learn from their students the importance of "playing around" with ideas and learning from their mistakes. These authors should be commended for pushing us into a new era of teaching Logo ... one that affects the total learning environment of the child.

Do you know of any new Logo materials and books? Did you find them useful in teaching? If so, we invite you to send information to The Logo Information for Teachers (LIFT) Group, PO Box 5396, Plymouth, MI 48170.

Anne Cairns Federlein is a professor in the Early Childhood Area, School of Human and Educational Services, Oakland University, Rochester, MI.

Q and A

by

Jim McCauley

(Ed. note: Jim McCauley is in the process of moving to Eugene, OR, where he will pursue his Ph.D. under the supervision of David Moursund. His regular NLX column will resume with the October issue. In the meantime, you may send your Logo questions to him at 55 Sunset Drive, Eugene, OR 97403.)

Logo in the United States: A Survey

by

Joyce Tobias

During the past winter, survey forms were sent to members of the National Logo Exchange and the Logo 84 conference participants to learn more about the Logo community. The results of only the 366 respondents from the United States have been included in this report.

As in the results of the survey conducted by the National Logo Exchange in 1984 (Ed. note: see NLX September 1984, p. 10, or "Gradations of Logo" in The Computing Teacher, December 1984 / January 1985, p. 56), a wide majority of respondents use some form of Logo on the Apple computer (Terrapin, Krell, Apple, or Sprite Logo). Both Commodore and IBM have shown considerable gains since last year. Logo use with the Texas Instruments and Coleco computers is expected to diminish since those computers are no longer being manufactured.

Computers Used With Logo

77%	Apple
7%	Commodore
7%	IBM
5%	Atari
2%	Texas Instruments
2%	Coleco

The table below summarizes the indications of where Logo is being used. Not surprisingly, the Logo use is concentrated in the elementary schools, both public and private, with universities / colleges, and high schools gaining from last year. Logo use in the homes is also gaining in significance. A growing use of Logo in clubs is evident.

Responses in the "Other" category below include programs for the gifted and talented, special needs programs, summer camps, adult education, recreation departments, church groups, museums, hospitals, rehabilitation centers, and correctional institutions.

Where Logo Is Used

42%	Elementary Schools
17%	University / College
16%	Homes
14 %	Secondary Schools
4%	Clubs
7%	Other

The table below indicates the response to the question, "How is Logo being used?" The "Other" category includes recreation, preservice and inservice training, logic, and artificial intelligence.

Logo Use

72%	Education
10%	Research
10%	Hobby
1%	Business
7%	Other

Of the respondents indicating educational use, 86% checked programming and 52% checked use as a tool to teach other subjects. (Many respondents checked both boxes.)

Subjects in which Logo is used as a tool included 62% math, 26% language arts, 20% problem solving, 11% critical thinking, 9% science, 8% social studies, 20% other (computer literacy, music, art, special education, and design). (Many respondents checked more than one subject.)

Some of the most difficult concepts to learn were reported to include list processing, recursion, integrating Logo into the curriculum, variables, file and workspace management, output, music, and language syntax.

When asked for the areas in which additional help was needed, the respondents included only list processing and integrating Logo into the curriculum from the difficult concepts above. Other topics included creating microworlds and applications, music, sprites, "everything!", and ideas for using Logo with adults.

Particularly helpful Logo materials cited included Learning with Logo (Dan Watt, McGraw-Hill), Apple Logo (Harold Abelson, McGraw-Hill), self made materials, and the National Logo Exchange newsletter. Others mentioned included Logo manuals, The Turtle's Sourcebook (Donna Bearden et al., Reston Publishing), Nudges (Steve Tipps et al., Holt, Rinehart and Winston), and Turtle Geometry (Harold Abelson and Andrea diSessa, MIT Press).

The survey raised a question for me. As I tabulated the responses, I noticed that many people mentioned "problem solving," others wrote down "critical thinking," and still others mentioned both. What is meant by these terms? How is Logo enhancing these skills? I would be interested in your comments and ideas on this topic. Please write to Joyce Tobias, 38 Maple Avenue, Lexington, MA 02173.

Joyce Tobias is an educational consultant and works with Microworlds Learning, Inc., in Cambridge, MA.

Logo 85 Report

The Logo 85 conference was held in late July at MIT. Copies of the Logo 85 Preproceedings and Theoretical Papers are available for $15 each. Send a check (payable to MIT Logo Group) to Logo 85 Preproceedings, MIT AMT, E15-309, 20 Ames Street, Cambridge, MA 02139.

At the conclusion of the conference, Seymour Papert suggested that reflection on the subjective side of Logo is important and has not yet found an adequate place in the literature. He asked that teachers and researchers using Logo collect anecdotes and incidents and send them to him (at the above address) for a study of the range and diversity of human experience. Letters, outlines, plans, free associations, all are welcome.

Here are some comments on the conference from those who attended.

Kathleen Nutting:

Logo 85 provided an exciting opportunity to learn from a diversity of individual perspectives about Logo. I discovered that Logo's richness supercedes any one language as I interacted in an environment of individuals from all cultures. The people truly furthered a learning experience not only with Logo, the language, but with the process of change and growth in education, with honesty and risks, with issues in teacher training, with encouraging cultures and microworlds, and with enabling learning to be a process.

Logo, much as with any language, is a process of combining disjointed segments, words, and commands, to create a picture of reality. Seymour Papert challenged us to move the turtle from the flat screen into a practical reality that applies to life situations. He spoke of using Logo as a dynamic tool to settle questions that arise in classrooms and using Lego™ motors and kits in conjuction with the computer and Logo to provide a discovery-rich environment. Students may be guided in duplicating realities such as circles, and in creating new realities through their own visions. Horacio Reggini demonstrated such a vision through his presentation of a three-dimensional reality.

Logo 85 provided me with a broader perspective of the possibilities with Logo, new issues to ponder, and ideas and insights to share.

Ihor Charischak:

Though Logo 85 was a three day event, it was actually a three week experience for me. During the two weeks prior to the conference, I had the opportunity to participate in Seymour Papert's improvisational "Samba School." He invited inexperienced teachers from a local school, veteran teacher trainers from various parts of the world, elementary school students, and MIT faculty and students.

My environment consisted of people who presented ideas formally, my colleagues and friends who discussed the ideas with me informally, and the MIT culture. I soon found myself thinking about what it takes to make learning happen in places called schools, and how Logo can make a difference. Each new idea gave me insights into a new way of looking at this old problem. It became exciting! I thought about what was happening to me. Why was I enjoying this so much? I delighted in the freedom! I felt like a pioneer exploring in the realm of ideas!

Many people came to Logo 85 looking for ways to calm the chaos in their schools and in their own minds. I heard someone say, "My kids love Logo, but what are they learning? They're having too much fun. We need more structure." Logo critics would applaud this. They want proof that Logo teaches this or that. They are overlooking Logo's real power: it can make children fall in love with learning. But how does one measure that?

I fell back in love with learning during my preconference time. But, did I become a better planner? Can I now do math faster? Am I smarter? Probably not. But I do know that I'm happier and more confident as a result of my experience. And who knows what I can accomplish when I feel this way?!

Paul Lorentzen and Arun Keshap:

Logo 85 was a veritable collage of excitement, a plethora of activity, uplifting for the newcomers and rejuvenating for the old. It was an excellent blend of national and international involvement.

It is a wonder that there are still academicians who do not realize Logo's multiplicity of uses. If they had attended the conference, they would have seen the range of Logo extend from physics and meteorological forecasting to adventure games and all levels of teaching. They would have enjoyed the open, casual atmosphere and the chance to share experiences with new-found friends. They might have even gotten a glimpse into a teaching method of the future - Lego™ and Logo, a combination of two simple but limitless building blocks of knowledge. To top it off, they could have mingled with the "visionaries" in the exhibition hall to discuss the latest Logo ideas and products.

Logo 85 was an enjoyable experience organized by remarkable people. They are to be commended for all events. We can't wait for Logo 86!

Gayle Lawrence:

It was evident from the conference that Logo is blooming at a time when educators worldwide are troubled by the scarcity of such stimilating environments for students. It is an unhappy realization that the educational community is failing to produce students who possess well developed problem solving and critical thinking skills which they can use in meaningful and productive ways.

As evidenced by presentations such as Ann Thompson's (of Iowa State University), Logo gives us a chance to set aside some of the low risk "pat answer" curriculum materials and to explore a world of dynamic, interactive, and (at times) unpredictable learning.

Molly Watt's demonstration of a model for recursion illustrated that teachers and learners alike are free to create their own models for the powerful ideas embedded in the Logo language. Dan Watt presented microworlds created by teachers using Logo, and suggested how these might be used to accomplish learning goals. Clearly, from David Thornburg's informative and entertaining presentation of the use of ExperLogo by Stanford University students, we are limited only by our own imaginations and creativity.

Kathleen Nutting is a Resource Teacher in the Computer Science Department of Jefferson County (CO) School District. Ihor Charischak conducts workshops for Logo Computer Systems, Inc. Paul Lorentzen is a student of nuclear engineering at the Georgia Institute of Technology. Arun Keshap is majoring in mechanical engineering at the University of Virginia. Both students use Logo in their studies.Gayle Lawrence is a computer teacher for grades 1 - 5 in Troy, NY.

NLXual Challenges

by Robs Muir

ENTOMOLOGY: The study of insects, i.e., bugs.

The word "entomology" is derived from a Greek root meaning to cut . The following NLXual Challenge constitutes an fascinating invitation from E. Paul Goldenberg to go on a "bug hunt" within an interesting Logo project. How good a Logo entomologist have you become? Can you "cut the bug?"

Bug Hunting

by E. Paul Goldenberg

Normally, I don't go out of my way to ask people not to be creative, but I have a rather subtle bug for you to find. This bug's residence is in a locale oft referred to as "the garden path." The flowers along this path are some of the sweetest and most seductive on the face of the globe.

To keep you moving, I must ask you to indulge me and allow me to lead you down the path. I promise that I will not mislead, but I cannot promise that I will point out where you may be stung.

In any event, stay close by me and, if you would like to see this most subtle of bugs, do not try any experiments I have not asked you to try just yet. Type my procedures exactly as I show them.

(If you have an MIT syntax Logo like Terrapin Logo, you will have to change the syntax of the lines containing IF and use a substitute for EMPTYP, but make no other changes for fear of shooing the bug away before you see it.)

TO BUG :WHERE

Here is a procedure that increments by putting into a box named :COUNTER the value of 1 plus whatever had been previously inside that box.

```
TO INCREMENT :COUNTER
MAKE :COUNTER (1 + THING :COUNTER)
END
```

Try it out this way.

```
PONS (to show all the names you have - type PO
NAMES with MIT Logos)
MAKE "MY.SCORE 0
MAKE "YOUR.SCORE 0
INCREMENT "MY.SCORE
PONS (to show how the names have changed)
INCREMENT "MY.SCORE
PONS
```

Now, using INCREMENT, create a procedure TALLY that counts how many times each letter occurs in a word.

```
TO TALLY :WORD
IF (EMPTYP :WORD) [STOP]
INCREMENT FIRST :WORD
TALLY BUTFIRST :WORD
END
```

To try out TALLY, you first need to initialize the counts of the letters you are about to count. So, try this sequence.

```
MAKE "A 0
MAKE "E 0
MAKE "I 0
MAKE "R 0
PONS   (you should have MY.SCORE being 2, and 5
other names all set to 0)
TALLY "AIRIER
PONS   (A and E should both be 1, while R and I should
both be 2.
TALLY "EERIER
PONS   (now the counts should reflect the totals in both
words)
```

That, of course, suggests the idea of counting the frequency of occurences of letters in an entire sentence by tallying each word sequentially.

```
 TO TALLYSENT :SE
 IF (EMPTYP :SE) [STOP]
TALLY FIRST :SE
TALLYSENT BUTFIRST :SE
END
```

Of course, we must go through a lot of initialization before using TALLYSENT each time, so we have now developed a need for an initializer. This procedure uses a subprocedure that follows the same kind of logic that TALLY and TALLYSENT followed.

```
TO INIT :L
IF EMPTYP :L [STOP]
 MAKE (FIRST :L) 0
INIT BUTFIRST :L
END
```

Aren't the flowers beginning to smell particularly sweet here? Now, we are quite close, so be very still and don't run ahead! Try INIT exactly as I tell you to!

NLXual Challenges continued

```
PONS
INIT [A B E G H I N R S T U W Y]
PONS
```

Look good? Yes, and can we tally a sentence that contains the right letters? Let's see.

```
TALLYSENT [BE WARY THE BUG IS NEARBY]
PONS
```

Ah, it works just as we wanted. We have caught three bees, but not a single bug yet. Before casting the net again, let us save ourselves some typing by creating an initializer for all the letters of the alphabet.

```
TO INITIALIZE
INIT [A B C D E F G H I J K L M N O P Q R S T U V W X Y Z]
END
```

Ah! There the bug is, resting almost invisibly on the petal of the nearest flower. Easy now. Slowly. Quietly. Sneak up on it and...

```
INITIALIZE
TALLYSENT [QUIETLY NOW YOU ARE
ALMOST UPON IT]
```

What happened?!

```
PONS
```

Did you catch the bug? Where is it now?

Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ...Δ

We invite you to send a short letter outlining your frustrating bug search along with a self addressed stamped envelope to: NLXual Challenges, Attn: Robs Muir, 1688 Denver Avenue,Claremont, CA 91711. In return, we will send you one possible approach to a solution. Happy hunting!

In addition, if you or a Logo colleague would like to suggest an NLXual Challenge for other NLX readers, send your problem to the above address.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.
E. Paul Goldenberg is a teacher at Lincoln-Sudbury Regional High School in Sudbury, MA.

Logo Notices

The first World Logo Conference is scheduled for October 25 - 27, 1985. Offered through CompuServe, the conference makes it possible for attendees to participate without leaving home. Online panel discussions with well known Logo workers will be featured, in addition to other activities. If you are already a CompuServe member, registration is $15. Otherwise, the cost is $55, and includes a CompuServe membership. It is necessary to use a modem to participate in the conference (and other CompuServe activities). For more information, write to World Logo Conference, 1616 Charles Street, Vancouver, BC Canada V5L 2T3.

"Logo Under Fire" will be the subject of a segment on "Educational Computing," a weekly half-hour program produced for public television by Kentucky Educational Television (KET), in which the pros, cons, and future of Logo will be examined. Scheduled for broadcast September 26 and 27, the Logo segment will include an interview with Judah Schwartz, Harvard professor and critic, along with footage taped at the Logo 85 conference at MIT. Educational Computing's commentator, Fred D'Ignazio, has promised a stimulating program. Consult your local public television station for local airdate and time.

For a free catalog describing Logo teaching materials, write to K-12 MicroMedia, Dept. LL, 172 Broadway, Woodcliff Lake, NJ 07675, or call (800) 922-0401.

The National Logo Exchange, copyright © 1985 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717.

Editor..........Tom Lough

THE NATIONAL
LOGO
EXCHANGE
The Logo Newsletter for Teachers

Multiple Turtles Without Sprites

by James Hurley

My interest in Logo is in its appplication as a tool in learning and teaching physics. Most problems in physics involve more than one object, and need Logo versions with multiple turtles for an effective representation.

Unfortunately, the most widely used Logos (for example, Apple Logo and Terrapin Logo) lack multiple turtles. However, there are ways to simulate the effects of multiple turtles using only a single turtle Logo version. In this article, I present one method I use with my physics classes.

Turtle Herd

Atari Logo has four turtles, and uses the ASK command to control them. If you type

```
ASK 1 [ HOME FORWARD 50 ]
```

turtle number 1 goes to the home position and heading, then goes forward 50 steps. A procedure which performs the same tasks as ASK could be written for single turtle Logos to simulate multiple turtles.

To accomplish this, we may use the turtle number as a variable, and associate with it a list of two elements: the list of the position, and the heading. For example, if we type

```
MAKE 1 [ [ 30 40 ] 90 ]
```

then turtle number 1 has associated with it a position of (30, 40) and a heading of 90°. Sometimes information such as that in this list is called the *state* of the turtle.

Now let's set up a herd of four turtles, numbered 0, 1, 2, and 3, to keep the "main" turtle company. The HOME.ALL procedure initializes the new turtles and places them all at HOME. (The procedures which follow are for LCSI Logos. See the notes at the end of the article for MIT Logos, such as Terrapin.)

```
TO HOME.ALL
LOCAL "N
MAKE "N 0
REPEAT 4 [ MAKE :N [ [ 0 0 ] 0 ] MAKE "N :N + 1 ]
END
```

If you type PONS, you find that all four turtles have the same state; that is, all are at (0,0) with a heading of 0°.

```
3 IS [ [ 0 0 ] 0 ]
2 IS [ [ 0 0 ] 0 ]
1 IS [ [ 0 0 ] 0 ]
0 IS [ [ 0 0 ] 0 ]
```

Now that we have initialized the turtles, we can use the information in the lists to manipulate the turtle states. For example, we can direct any of the turtles with ASK.

```
TO ASK :TURTLE.NUMBER  :COMMAND.LIST
MAKE "OLD.POS.HEADING LIST POS HEADING
PU SETPOS FIRST THING :TURTLE.NUMBER
SETHEADING LAST THING :TURTLE.NUMBER
PD RUN :COMMAND.LIST PU
MAKE :TURTLE.NUMBER LIST POS HEADING
SETPOS FIRST :OLD.POS.HEADING
SETHEADING LAST :OLD.POS.HEADING PD
END
```

Multiple Turtles continued

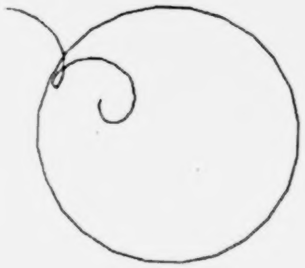
In ASK, we first note the state of the "main" turtle and record this as the OLD.POS.HEADING. Next, we set the ASKed turtle to its former position and heading, and run the COMMAND.LIST. After this action, the turtle may be in a different state, with a new position and heading. So the turtle state list is updated by MAKE :TURTLE.NUMBER LIST POS HEADING. The "main" turtle is then put back in its original state, ready to continue.

All instructions in the COMMAND.LIST must be commands. If an operation is included, a command must be present to accept the input. Otherwise, the ASK procedure stops.

The lists which were set up with HOME.ALL could be extended to include the pencolor, whether the ASKed turtle is shown or not, and other characteristics. Some students like to use names instead of numbers to identify the ASKed turtles. What other extensions can you think of?

Follow That Turtle

Here is a simple demonstration of an activity which draws the path of a homing turtle following a target turtle traveling in a circle. Part of the homing path is shown in the figure. What do you think the rest of the path will look like?



```
TO SETUP
HOME.ALL
ASK 0 [ PU SETPOS [ 80 80 ] SETHEADING 120 PD ]
ASK 1 [ PU SETPOS [ -80 80 ] PD ]
FOLLOW
END
```

```
TO FOLLOW
ASK 0 [ FD 25 RT 15 MAKE "TARGET POS ]
ASK 1 [ SETHEADING TOWARDS :TARGET FD 5 ]
FOLLOW
END
```

You can vary the results by changing the inputs to FD in FOLLOW. These inputs correspond to the speeds of the two turtles.

Hunt the Monkey

Here is a more complex application of multiple turtles I use in my physics classes. This simulates a classic demonstration called "The Monkey and the Hunter."

A monkey is hanging from the limb of a tree. A hunter some distance away aims a rifle *directly* at the monkey and fires. At the same instant, the monkey lets go of the limb and drops, hoping to get out of the way of the bullet. Unfortunately for the monkey, the bullet finds its mark ... *every time!*

The reason for this remarkable result is that the bullet follows a parabolic trajectory whose shape depends on the muzzle velocity of the bullet. If the velocity is high, the bullet travels quickly and strikes the monkey at a position just below the limb, before it has fallen very far. If the velocity is low, the bullet hits the monkey after it has fallen some distance.

The purpose of this experiment is not frivolous. It demonstrates Einstein's principle of equivalence: the equivalence of an accelerated frame of reference and a gravitational field. If the experiment is viewed from the point of view of an observer falling in the gravitational field with the monkey, the monkey is seen at rest, with the bullet traveling in an apparent line straight to the mark.

Here is a set of procedures for this experiment.

```
TO MONKEY.AND.HUNTER :MUZZLE.VEL
        :MONKEY.POS
CS FULLSCREEN
HOME.ALL
MAKE "GRAVITY 0.2
MAKE "MONKEY :MONKEY.POS
MAKE "HUNTER [ -100 -100 ]
HIDETURTLE
ASK 0 [ PU SETPOS :HUNTER SETH TOWARDS
        :MONKEY MAKE "DIRECTION.TO.MONKEY
        HEADING ]
ASK 1 [ PU SETPOS :MONKEY ]
STEP (:MUZZLE.VEL * SIN :DIRECTION.TO.MONKEY)
        (:MUZZLE.VEL * COS
        :DIRECTION.TO.MONKEY) 0
END
```

```
TO STEP :VX :VY :VEL.MONKEY
ASK 0 [ INC.XY :VX :VY ]
ASK 1 [ INC.XY 0 :VEL.MONKEY ]
STEP :VX (:VY - :GRAVITY) (:VEL.MONKEY -
        :GRAVITY)
END
```

```
TO INC.XY :X :Y
SETPOS LIST (XCOR + :X) (YCOR + :Y)
END
```

NLXtra

A Supplement to the National Logo Exchange Newsletter

Logo Notices

The Young Peoples' Logo Association has gone electronic! If you are a member of CompuServe, be sure to visit the Logo Forum, a veritable electric Logo kingdom, with lots of goodies and treasures scattered about! From any ! prompt, type GO LOGOFORUM. If you have been looking for a reason to join CompuServe, here it is.

Ideas y Formas, a Logo book by Horacio Reggini published in Spanish, explores the concepts of shapes in space: their description, generation, and manipulation. For more information, write to Emece Distribuidora, Alsina 2062, 1090 Buenos Aires, Argentina.

Celia Hoyles, of the London University Logo Group, announces the availabilty of the following publications on Logo. (All prices are listed in English pounds due to the uncertainty of the monetary exchange rates.) *The Interim Report on the Logo Mathematics Project 1983-4*, £8.50 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Proceedings of the Logo and Mathematics Education Conference, March 1985*, £5.00 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Creating a Mathematical Environment through Programming: A Study of Young Children Learning Logo*, by Richard Noss, £10.00 plus postage and packing: £10.00 air mail or £5.00 surface mail.

Computer Technology for the Handicapped, a conference sponsored by Closing the Gap, will be held October 30 - November 2, at the Radisson South Hotel in Minneapolis, MN. Pre-conference workshops, over 100 presentations (including several on Logo), and a wide range of exhibits are scheduled. For more information, write to Closing the Gap, PO Box 68, Henderson, MN 56044.

The World Congress of Education and Technology will be held in Vancouver, British Columbia, May 22 - 25, 1986. The program contains a "who's who" of educational computing, and promises many exciting Logo events for the expected 10,000 attendees. Make plans now to attend this global event! Write to Anne Marx, WCET, 1155 West 8th Avenue, Vancouver, BC V6H 1C5 Canada, or call (604) 734-2721.

The sixth annual *Microcomputers in Education Conference* is scheduled for March 12 - 14, 1986, at Arizona State University in Tempe, AZ. The theme is "Ethics and Excellence." For more information, write Janie Hydrick, Payne 216, College of Education, Arizona State University, Tempe, AZ 85287, or call (602) 965-7363.

The *National Educational Computing Conference* (NECC 86) is scheduled for San Diego, June 4 - 6, 1986. Hosted by the University of San Diego, NECC 86 will offer Logo users many opportunities for meeting and sharing. Program proposals and papers are accepted until November 1, 1985. For more information, write to NECC 86, University of San Diego, School of Education, Alcala Park, San Diego, CA 92110.

The Journal of Computers in Math & Science Teaching (JCMST) is beginning its fourth year of publication, and now offers an expanded format to its subscribers. A regular Logo column is included, along with articles, reviews, listings, and announcements. The JCMST is published quarterly at $18, and is devoted to the use of computers in mathematics and science teaching at all levels. For more information, contact Mary Marks, JCMST, PO Box 4455, Austin, TX 78765. (512) 244-1771.

Computers in the Schools has published a special double issue, *Logo in the Schools*. The 300+ page volume contains 30 articles by well known Logo workers, including a feature article by Seymour Papert. This highly recommended volume is available for classroom adoption at $7.95 per copy. A minimum order of 5 copies is required for this special price. Haworth Press, 28 East 22nd Street, New York, NY 10010-6110.

The outstanding NOVA public television program, "Talking Turtle," is now being distributed in 16mm film and video cassette formats by The Media Guild. An overview of Logo, an interview with Seymour Papert, and Logo applications in a variety of settings are featured in the 48-minute film. For more information, write The Media Guild, 11526 Sorrento Valley Road, Suite J, San Diego, CA 92121, or call (619) 755-9191. (Ed. note: See the December 1983 NLX for a review by Andrew David.)

The Atari Teachers' Network publishes a quarterly newsletter with loads of tips and information. A recent issue contained a review of PRINTWIZ (Allen Macroware, PO Box 2205, Redondo Beach, CA 90278, $29.95), a disk based graphics screen dump which works with the Atari Logo cartridge and a number of different printers. To subscribe, send $4 to John Hanna, Editor, Atari Teachers' Network, Teaneck High School, 100 Elizabeth Avenue, Teaneck, NJ 07666.

NLX reader John L. Earnest of Garden City, NY, reports an unexpected result when using an input value of 1.5 with the ARCTAN operation of Apple Logo.

```
PRINT ARCTAN 1.4
54.4626
PRINT ARCTAN 1.5
62.9897
PRINT ARCTAN 1.6
57.9953
```

A further curiosity is as follows.

```
PRINT ARCTAN (1.0 + 0.5)
62.9897
PRINT ARCTAN (1.4 + 0.1)
56.3105 (the correct value!)
```

Be aware that this bug exists, and that Apple Logo calculations using ARCTAN with an input near 1.5 may be subject to error. ARCTAN in Apple Logo II and Terrapin Logo appears to work properly in this range.

Students of Robin Conrad, a teacher in New York City, use SETSCRUNCH with their NLX ABC's to vary the proportions of the letters. Robin also suggests using negative numbers as inputs for the letter sizes to explore symmetry in a surprising way. For a free listing of the procedures to draw the letters of the alphabet, send a self-addressed stamped envelope to: NLX ABC's, PO Box 5341, Charlottesville, VA 22905.

A number of Spanish Logo versions are available from Chile. For information on Spanish Logo for the Apple and Atari, write MDS, S.A Victoria Subercaseaux 191, Dpto. 201, Santiago, Chile. A Spanish Logo for the Commodore is available from Andromeda Computacion Ltds., Manuel Montt, 037 Local 10, Santiago, Chile.

ExperTelligence announces ExperLogo 1.1, an upgrade to their ExperLogo 1.0 with more features and some bug fixes. Additional features include user definable menus and window manipulation from program control and the inclusion of ERALL and CLEARTEXT commands. The company has also announced ExperLogo-Talk, a program written in ExperLogo which adds *speech* to the Logo capabilities of the Macintosh. It requires 512K and an external drive, and sells for $49.95. For more information, write to ExperTelligence, 559 San Ysidro Road, Santa Barbara, CA 93108, or call (800) 969-7874.

Logo users in central New York state are invited to join ASTROLUG, the Albany - Schenectady - TROy Logo Users' Group. For more information, write to Peter Rawitsch, PO Box 254, Guilderland Center, NY 12085, or call (518) 861-6637.

BIKILOG is short for Big Kids Logo Group, an organization formed to give Australian Logo users of all ages an opportunity to meet and interact. For more information, write BIKILOG, Box 88, Balaclava 3183, Australia.

Chris Templar would like information from anyone on techniques to move Commodore Microillustrator files into Commodore Logo. If you know a method to do this, please correspond with Chris Templar, Department of Computer Education, Johnson Bible College, Knoxville, TN 37998.

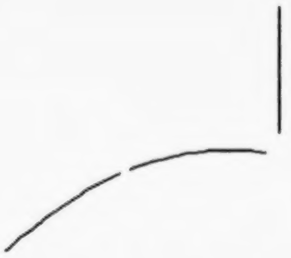
A free author's guide to writing articles for the NLX is available. To obtain your copy, send a self addressed stamped envelope to NLX Author's Guide, PO Box 5341, Charlottesville, VA 22905.

Learning Ukranian with Logo is a software development in Sprite Logo by Ihor Charischak. If you are interested in participating or would like more information, send for a free brochure. Ihor Charischak, Logo Computer Systems, Inc., 555 West 57th Street, Suite 1236, New York, NY 10019. (212) 765-4780.

The main procedure, MONKEY.AND.HUNTER, has two inputs. The first input lets you choose the muzzle velocity of the bullet. The second input is the initial position of the monkey. If you type

```
MONKEY.AND.HUNTER 8 [100 80]
```

and humanely interrupt the program before violence is done, the figure below is produced.



If you have a Logo version with multiple turtles already on board (Atari Logo, Commodore Logo, Sprite Logo, etc.), you can begin to program these applications directly.

There are many other uses for multiple turtles, of course. I have used as many as 100 of them stationed along the x-axis to draw a histogram, for example.

Although the ASK procedure is not particularly fast, it does the job and demonstrates once again the pleasure of working with an extensible language. As you and your students gain more experience with Logo, the power of this extensibility will make itself known.

MIT Logo Notes

The MIT Logos do not use the same approach to position as LCSI. Instead of POSition lists and the SETPOS command, the x- and y-coordinates are used separately. I suggest that the following HOME.ALL procedure be used to develop a group of MIT Logo procedures similar to those in this article.

```
TO HOME.ALL
LOCAL "N
MAKE "N 0
REPEAT 4 [ MAKE WORD "  :N [ 0 0 0 ] MAKE
      "N :N+1 ]
END
```

James Hurley uses Logo to teach physics at the University of California at Davis, and is the author of *Logo Physics*, published by Holt, Rinehart and Winston.

From The Editor

by Tom Lough

Names are fascinating. I marvel that we humans attach names to practically everything, from the muscles and bones of the body to the emotions dominant at various stages of growth and development, from the characteristics of the smallest component of the subatomic world to the descriptors of the universe itself.

We also attach names to ourselves. When Posy and I got the news that our son was on the way, we knew we faced the somewhat awesome responsibility of assigning a name to a fellow human being. After reading through a number of "Name Your Baby" books, I wrote a Logo program which used a list of 550 first names, and printed an endless list of various combinations of them with "Lough."

There were some interesting ones, to be sure. Those of you who saw the printout at MECC 82 might remember the likes of Winthrop Maximilian Lough, Chad Leo Lough, and Kyle Sherlock Lough!

But we didn't use these names, because they had no meaning for us. Instead, we chose Kyser Thomas Lough, which contains an old family name and the name of his dad (that's me!).

When we call Kyser by his name, we are not just using a word which has been assigned as his label. We are using a word that means something to us. (By the way, he is into his "terrible twos," and we have to call him quite often for various reasons!)

At a workshop this summer, Seymour Papert challenged us to concentrate on the NAME primitive and the thinking which is associated with it.

Using meaningful names is certainly part of the process. But, in order to NAME a variable or a procedure meaningfully, we must understand something about what it is supposed to represent or do. The search for this understanding is a very special kind of learning that sticks to one's ribs, so to speak.

Let's invite our students to play the name game at deeper levels, by encouraging them to choose descriptive words based on a personal understanding they have discovered for themselves. After all, isn't this what taking responsibility for one's own learning is all about?

The next time someone calls you by name, let it serve as a reminder of the process which accompanies meaningful naming and affirm your determination to explore more of the ideas in names.

I suppose Glen would call it something like "Nameful Thinking."

FD 100!

Tom

Tipps for Teachers

by Steve Tipps

Rooms for Logo

Look around the room where you are reading. You probably see chairs and tables and windows. Whether you are at home or at school, certain objects are in your view. For each of you, the arrangement is different. If you leave the room and sketch all the things you can remember, the sketch is an approximate reconstruction of your room. Your sketch can also be used as a kind of rudimentary map of your territory.

Globes and mercator projections represent the entire Earth's surface. Road maps are needed to plan and to complete vacations. City maps are used to locate homes, parks and restaurants. In each case, the map is a symbolic picture. Success with maps presumes that children (and adults) have concrete ideas for the symbols.

Map making is a good way to make the connection between real objects and symbols. Logo is a good tool for map making. Turtle graphics can be used to draw the representations. The extensibility of Logo allows maps to be adapted to each class or child.

Class Maps

Map skills are included in most elementary social studies programs. While some programs begin with formal maps, most are more sensitive to the cognitive understanding of children. The latter emphasize maps of the child's environment--classroom, school, home, neighborhood. These are close to the child and their features are already familiar.

In a classroom, for instance, the objects in the class can be listed: desks, chairs, teacher's desk, cabinet, bookshelves, tables, rugs, fish bowl, sink. Making a list of things in the room is a good cataloging exercise for kindergarten and first grade children. Labels can then be attached to many of these objects. Labelling or naming is a powerful idea with or without Logo. The list of things in the classroom could grow as children make greater and greater differentiation of objects: reading table, science table, art table.

Making a model of the room requires a different kind of problem solving. The teacher can vary the amount and kind of assistance given as well as the kind of model being worked on. The most concrete model would use toy furniture. Since few classes have toy furniture, wooden blocks could be used for a model. The wooden blocks might be borrowed from the kindergarten block corner or from wood scraps. (Beware of splinters in rough wood however.)

A suitable "floor" to work on is something the teacher may want to provide. A floor could be a shallow box such as holds large drawing paper, a piece of paper with a room outline on it, a cafeteria tray, or a table top.

Model Rooms

A wooden model requires several skills. *Size discrimination* is needed to see that the teacher's desk is bigger than the children's desks or tables. Whether exact measurement is important depends on the objectives the teacher has in mind for the project. Some children even at first grade may want the teacher's desk to be twice as big, while other are content with a big block and a small block.

Another skill which is part of model building is a *change of perspective*. Rather than being at the same level as the objects in the class, the student is "above" the objects in the class. "Bird's-eye view" is a difficult but necessary map skill.

Orientation is another spatial skill involved in map making. Which side is the front? Where are the windows and doors? Should be floor be turned so that the map is oriented the same way as the class? While some children may be able to rotate objects 90 degrees or 180 degrees, most need to have the floor pointing the same way as the actual room. Being able to rotate objects in space requires imagery which does not develop until upper elementary grades. The model might be placed in the classroom so that the entire room can be seen while children work on it.

Children may want to tape *labels* on the blocks with names of teacher, student or class pet: Mr. Roarke, Melissa, Fluffy. Now instead of all the desk blocks being the same, they represent discrete places in the room. Order is important. Names have meanings of possession and position.

Tipps continued

Floor Plans

A different level of abstraction is a paper model of the class. Paper squares and rectangles are two dimensional pictures, rather than three dimensional representations, A desk-sized classroom map is made with moveable paper pieces for furniture. One piece of paper could be the floor, while another could be cut up for objects. After work with the wooden model, the students should have fewer problems with this level of map making.

Exact scale maps are not the objective for younger children. By fourth or fifth grade, scale may be important. Then, teacher and children should engage in a measurement project and decide together what size the floor and desks should be on paper. A classroom template for the map might be developed and duplicated.

Turtle Maps

The map activities described are good activities for any classroom. Logo is another tool which can extend and enhance children's facility with map making. Decisions about how best to use Logo as a map making tool depend on the teacher's objectives, student skills, and time available for computer work.

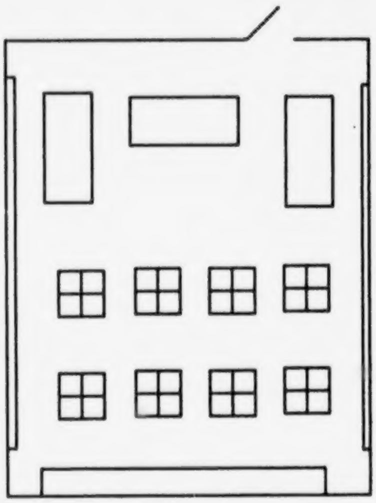
The most wide-open use of Logo would simply be a challenge posed to have students make a map on the screen like the wooden or paper models. Children would need to be able to create procedures for squares and rectangles to attack this problem with success. The list of objects in the room provides the procedure names. One benefit of the list would be emphasize the modular nature of procedures. DESK might be used to draw all the desks in the room. TABLE would also useful for several different shapes.

Positional procedures such as MOVE would also be very helpful. Drawing an arrangement of desks in clusters or rows is much easier with a consistent way to get the turtle over to the next starting place. Children often have a difficult time with changes in location, or state. Understanding the importance of starting and stopping in the same place, or knowing where you are, is more difficult than an actual drawing which can be seen.

Understanding the problems which are likely to be encountered in drawing Logo maps helps the teacher prepare for questions. How should I show doors and window? How big in turtle steps should the desk be? How can I make a row of six desks? Referring to the wooden and paper models is the best answer to most of the questions. Other solutions may involve some cooperative work to write a MOVE procedure or variable procedures.

A different approach would be to provide several Logo procedures for map making tools. Instead of having children write procedures for DESK, MOVE, or ROOM.OUTLINE, the teacher could save a file with these rudimentary procedures already written. The objective in this case moves away from general problem solving and toward map making specifically. The problem for children is not how to make the procedures but how to combine them into an abstract map of the room. The wooden or paper models are still essential experiences to bring to the task.

One classroom map might look like this:



More Room

Even with procedures already written, children have a lot to do to put the pieces together. The basic map can be embellished with many special touches: a fish bowl on the table, an apple on the teacher's desk, the location of the flag. A class project might involve each child drawing a picture of a room in their house or apartment. Then the class could have a tour of rooms or a tour of school with each teacher's room represented. You might get an architect to come and show blueprints for their buildings. Several pieces of software are now available for professional and amateur home planners.

Whether you choose the pure Logo route of having students make maps from scratch or the quasi-Logo method of providing tools, students have a lot of freedom to add to and adapt the basic map In either case, you are making room for a lot of learning with Logo

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX He conducts Logo workshops for school systems throughout the United States.

Logo Teacher Feature

by

Rebecca Poplin

Featuring: Leslie Thyberg

"Ask three before me" has become one of the most quoted axioms in Logo teacher lore. With this statement, Leslie Thyberg told her children that they could solve their own problems by working together. (Ed. note: See "Too Many Questions, Not Enough Me," in the May 1985 NLX.) Her confidence in children and enthusiasm about Logo are obvious as she describes her work at the Learning Research and Development Center at the University of Pittsburgh. The school serves as a demonstration site for university classes and as a training facility for educators from other school districts.

Teachers at the Center are creating an adaptive learning environment model. Students in Leslie's K-3 class live in different areas of Pittsburgh and represent a broad spectrum of academic abilities, learning problems, and ethnic/social backgrounds.

During the past school year, excitement was generated by the addition of 15 PCjr's. These IBM computers were a result of Leslie's work with the National Computer Training Institute. Learning a new version of Logo with new features sparked enthusiasm. Music became much more prevalent in the children's work. Students created elaborate versions of "Raiders of the Lost Ark", various Bach tunes, and "The Star Spangled Banner." Creative writing and dramatics blossomed as well. Two girls even choreographed a dance using Logo.

Students were increasingly active as peer tutors. Some who already knew Terrapin Logo acted as dialect translators when IBM Logo came on the scene. These students worked after school in special seminars with the teachers to become turtle trainers. One pointer the turtle trainers were given was: " If you are tempted to touch the keyboard when you are helping another student, sit on your hands." The tutors became especially good at leading others to figure out the solutions for themselves rather than giving the solution. By the end of the year, all students had become turtle trainers and visited another school to teach the students there.

Not only did the students teach other students but the class had parent drop-in days on Fridays. Students taught their own lessons to their parents. Helping parents become familiar with computing and with Logo has been so beneficial that Leslie plans to expand the program this year.

With a major emphasis on integration of computing with other classroom activities, Leslie sets the themes and suggests possible tasks to complete, but she leaves the method to the student. This approach encourages multi-disciplinary projects which are completed at different levels. Squares might be a theme for the class. In the art center, students draw pictures using squares. In writing, they compose stories and poems about squares. On the computer, they create a square project which is either quite simple or very intricate, depending on the student's level. For example, a kindergarten student could make the square with Instant Logo, while a third grader might make a square design using variables.

Many teachers, like Leslie, are seeking ways of using Logo as an interdisciplinary tool. If you have ideas or know a teacher whose ideas would be of interest to others, please contact Rebecca Poplin, 2421 Fain Street, Wichita Falls, Texas 76308.

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

NLXionary

A Lectionary of Selected Logo Readings

with Commentary and Opinion

by Griff Wigley

"Logo Today: Vision and Reality," by Uri Leron, *The Computing Teacher*, February 1985.

>This is an important article for theorists and practitioners alike. The author provides a summary of four years' work using Logo with 6th grade students. More importantly, he seems to be one of those rare people who is able to speak knowledgeably about Logo's limitations and difficulties, yet remain enthusiastic about its promise.

>Leron says that the ideal of learning powerful ideas through a Piagetian style of learning poses a paradox in actual practice. If a totally non-directive approach is taken, many children do not acquire the powerful ideas available through Logo. If the powerful ideas are focused on too heavily, alienation and feelings of failure develop. The challenge is to find "more active ways of helping children without sacrificing the spirit of meaningful and exploratory learning." He calls his approach to resolving this tension "quasi-Piagetian learning" (QPL).

NLXionary continued

>He believes this additional structure is needed because he rarely sees children possessing what Papert describes as the burning curiosity and need to understand particular phenomena such as bugs in their programs. For example, they are content to fix bugs without understanding them; they are content to copy procedures without little or no desire to understand how the procedure works. He wonders if children are innately curious and that this is somehow lost over the years in a typical school . (I would argue that the preoccupation with achievement and academic performance as measured by tests are major factors in obliterating children's natural curiosity.)

>Leron describes the use of *QPL* in detail, focusing on the problems and possiblel approaches to teaching subprocedures, modularity, and debugging. He proposes further development of study guides that structure students' learning while preserving the spirit of exploratory and personally meaningful learning.

>This fellow clearly has a solid grasp of the issues facing Logo today. If your curiosity is still alive and you are sincerely interested in understanding and fixing the "bugs" in your Logo teaching approach, get this article and read it soon. I apologize for not alerting you to it sooner.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Logo Disserts:

Dissertations Dealing with Logo

by Barbara Elias

Children encounter a number of bugs as they learn to program in Logo. Removing these bugs (debugging) is an important programming step because it provides the opportunity for a variety of strategies to be used in clearing up errors. An investigation of student debugging was undertaken in an exploratory study involving 20 sixth grade children. The main purposes of the study were to identify and classify the bugs students encountered, identify the strategies they used, and describe their attitudes toward bugs.

Children who used a variety of programming approaches were selected by their teachers to participate. They were 11 to 12 years old and came from three elementary schools of the Edmonton (Alberta) Public Schools. The sample included 9 girls and 11 boys with limited Logo experience. Of the group, 9 were involved in their first year of Logo and 6 had had Logo prior to sixth grade. Following a pilot study, each student was observed while working on programming assignments about four times. The average observation time was two hours.

The study was conducted in 3 different and diverse classroom settings. One classrom had two computers available. Students in the second classroom had access to four or five computers in a room adjoining their classroom and those in the third school used five comptuers in a computer lab.

Programming assignments were provided by the individual teacher in each of the three schools. These incorporated a variety of Logo concepts: procedures, variables, recursion, conditional statements, and many more. Assignments were open ended but varied in the amount of structure. All allowed for discovery, exploration and creativity.

Individual observations, programs on disks, and log notes as well as tape recorded interviews were sources of data. Students were asked to "think aloud" and describe their debugging strategies as they worked. Each completed an attitude questionnaire.

The results indicate that 528 bugs were recorded and were classified into 8 categories. More than half of the bugs involved dimension (31%) or lateralization (18%) errors. Omission (14%) and orientation (13%) bugs were also recorded. More than 1200 strategies were identified. They were grouped into 14 areas with guess and check (41%), and observe and change (12%) having the highest percentages. Some of the other strategies included immediate mode (9%), clear screen and start again (8%), and physical referent (7%). Often, multiple strategies were used with the same bug. The students showed little evidence of detailed prior planning but exhibitied strong task commitment and positive attitudes.

The researcher recommends that teachers help children develop new debugging strategies and learn to use those already available to them more effectively.

>>>Cathcart, Gloria M. *Logo Programming Bugs and Debugging Strategies of Grade Six Students.* (Unpublished thesis, Master of Education, University of Alberta, 1985.)

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of Virginia.

Listful Thinking

by Glen Bull and Paula Cochran

LOGO WORD CHEMISTRY

A chemistry lab can be a fascinating place for young scientists. Test tubes and chemicals allow student investigators to explore laws of nature, and verify that the information in the book is really true.

On the other hand, spelling is rarely anyone's favorite subject. In contrast with the laws of nature that govern chemistry, the rules of spelling appear arbitrary and capricious. Perhaps that's why engineers are notoriously poor spellers. Schools have places where students experiment with chemistry laws, but there's no similar place where students can go to experiment with spelling rules.

In this column, we take a lesson from the chemists and set up a Logo Language Laboratory in which students can experiment with spelling rules and other linguistic laws.

Word Chemistry 101

Strange as it may seem, there is a kind of chemistry in the way words fit together, both for English and for other languages as well. Some sound combinations mix well together, while others don't ... just as different combinations of chemicals produce different results. For example, normally a plural in English is formed by adding *s*. However, if a word ends in *s*, *z*, *sh*, *sh*, or *ch*, you add *es*. *Mat* becomes *mats*, but *mess* becomes *messes*.

The reason that *es* is added becomes clear if you try to pronounce *messs*. It's impossible to pronounce unless you are a member of the Snake People of the Amazon. An English speaker can't say *messs* with a straight face. The spelling rule makes more sense if you understand a principle which underlies it.

Stocking the Laboratory

Linguists construct laboratories in their minds which allow them to experiment with the chemistry of words. In this column, we are going to use Logo to create a similar laboratory for students to use.

A chemistry lab is stocked with chemicals and equipment. We are going to stock our laboratory with the components that make up words. Linguists have categories of speech sounds like chemists have a periodic table which organizes the different families of elements. Some of the names linguists use might seem strange to a chemist, but on the other hand, some of the names chemists uses undoubtedly seem strange to linguists. PLOSIVE is the linguistic name for the type of sound that explodes with a burst of air. We have a Logo procedure containing the plosive sounds in English:

```
TO PLOSIVE
OP PICK [ P T K B D G ]
END
```

(See the end of the column for the PICK procedure listing.)

```
?PRINT PLOSIVE
K
```

Designing Experiments for the Lab

It's unlikely that you would just turn a student loose in the chemistry laboratory to randomly mix various combinations of chemicals together. Instead, projects and experiments are suggested that guide the student's work, and increase the probability that interesting results will occur. (Ed. Note: Also see the NLXionary elsewhere in this issue.)

In the same way, it is necessary to design some experiments for our linguistic lab. The experiment shown below asks the student to use Logo to put together various combinations of sounds that might make up a word.

You will have to type in the procedure LIQUID to do the following example. In addition to PLOSIVE and LIQUID, procedures for other sound categories can also be found at the end of the column.

```
?PRINT ( WORD "S PLOSIVE LIQUID )
STR
```

NLXtra
A Supplement to the National Logo Exchange Newsletter

Logo Notices

The Young Peoples' Logo Association has gone electronic! If you are a member of CompuServe, be sure to visit the Logo Forum, a veritable electric Logo kingdom, with lots of goodies and treasures scattered about! From any ! prompt, type GO LOGOFORUM. If you have been looking for a reason to join CompuServe, here it is.

Ideas y Formas, a Logo book by Horacio Reggini published in Spanish, explores the concepts of shapes in space: their description, generation, and manipulation. For more information, write to Emece Distribuidora, Alsina 2062, 1090 Buenos Aires, Argentina.

Celia Hoyles, of the London University Logo Group, announces the availabilty of the following publications on Logo. (All prices are listed in English pounds due to the uncertainty of the monetary exchange rates.) *The Interim Report on the Logo Mathematics Project 1983-4*, £8.50 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Proceedings of the Logo and Mathematics Education Conference, March 1985*, £5.00 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Creating a Mathematical Environment through Programming: A Study of Young Children Learning Logo*, by Richard Noss, £10.00 plus postage and packing: £10.00 air mail or £5.00 surface mail.

Computer Technology for the Handicapped, a conference sponsored by Closing the Gap, will be held October 30 - November 2, at the Radisson South Hotel in Minneapolis, MN. Pre-conference workshops, over 100 presentations (including several on Logo), and a wide range of exhibits are scheduled. For more information, write to Closing the Gap, PO Box 68, Henderson, MN 56044.

The World Congress of Education and Technology will be held in Vancouver, British Columbia, May 22 - 25, 1986. The program contains a "who's who" of educational computing, and promises many exciting Logo events for the expected 10,000 attendees. Make plans now to attend this global event! Write to Anne Marx, WCET, 1155 West 8th Avenue, Vancouver, BC V6H 1C5 Canada, or call (604) 734-2721.

The sixth annual *Microcomputers in Education Conference* is scheduled for March 12 - 14, 1986, at Arizona State University in Tempe, AZ. The theme is "Ethics and Excellence." For more information, write Janie Hydrick, Payne 216, College of Education, Arizona State University, Tempe, AZ 85287, or call (602) 965-7363.

The *National Educational Computing Conference* (NECC 86) is scheduled for San Diego, June 4 - 6, 1986. Hosted by the University of San Diego, NECC 86 will offer Logo users many opportunities for meeting and sharing. Program proposals and papers are accepted until November 1, 1985. For more information, write to NECC 86, University of San Diego, School of Education, Alcala Park, San Diego, CA 92110.

The Journal of Computers in Math & Science Teaching (JCMST) is beginning its fourth year of publication, and now offers an expanded format to its subscribers. A regular Logo column is included, along with articles, reviews, listings, and announcements. The JCMST is published quarterly at $18, and is devoted to the use of computers in mathematics and science teaching at all levels. For more information, contact Mary Marks, JCMST, PO Box 4455, Austin, TX 78765. (512) 244-1771.

Computers in the Schools has published a special double issue, *Logo in the Schools*. The 300+ page volume contains 30 articles by well known Logo workers, including a feature article by Seymour Papert. This highly recommended volume is available for classroom adoption at $7.95 per copy. A minimum order of 5 copies is required for this special price. Haworth Press, 28 East 22nd Street, New York, NY 10010-6110.

The outstanding NOVA public television program, "Talking Turtle," is now being distributed in 16mm film and video cassette formats by The Media Guild. An overview of Logo, an interview with Seymour Papert, and Logo applications in a variety of settings are featured in the 48-minute film. For more information, write The Media Guild, 11526 Sorrento Valley Road, Suite J, San Diego, CA 92121, or call (619) 755-9191. (Ed. note: See the December 1983 NLX for a review by Andrew David.)

The Atari Teachers' Network publishes a quarterly newsletter with loads of tips and information. A recent issue contained a review of PRINTWIZ (Allen Macroware, PO Box 2205, Redondo Beach, CA 90278, $29.95), a disk based graphics screen dump which works with the Atari Logo cartridge and a number of different printers. To subscribe, send $4 to John Hanna, Editor, Atari Teachers' Network, Teaneck High School, 100 Elizabeth Avenue, Teaneck, NJ 07666.

NLX reader John L. Earnest of Garden City, NY, reports an unexpected result when using an input value of 1.5 with the ARCTAN operation of Apple Logo.

```
PRINT ARCTAN 1.4
54.4626
PRINT ARCTAN 1.5
62.9897
PRINT ARCTAN 1.6
57.9953
```

A further curiosity is as follows.

```
PRINT ARCTAN (1.0 + 0.5)
62.9897
PRINT ARCTAN (1.4 + 0.1)
56.3105 (the correct value!)
```

Be aware that this bug exists, and that Apple Logo calculations using ARCTAN with an input near 1.5 may be subject to error. ARCTAN in Apple Logo II and Terrapin Logo appears to work properly in this range.

Students of Robin Conrad, a teacher in New York City, use SETSCRUNCH with their NLX ABC's to vary the proportions of the letters. Robin also suggests using negative numbers as inputs for the letter sizes to explore symmetry in a surprising way. For a free listing of the procedures to draw the letters of the alphabet, send a self-addressed stamped envelope to: NLX ABC's, PO Box 5341, Charlottesville, VA 22905.

A number of Spanish Logo versions are available from Chile. For information on Spanish Logo for the Apple and Atari, write MDS, S.A Victoria Subercaseaux 191, Dptc. 201, Santiago, Chile. A Spanish Logo for the Commodore is available from Andromeda Computacion Ltds., Manuel Montt, 037 Local 10, Santiago, Chile.

ExperTelligence announces ExperLogo 1.1, an upgrade to their ExperLogo 1.0 with more features and some bug fixes. Additional features include user definable menus and window manipulation from program control and the inclusion of ERALL and CLEARTEXT commands. The company has also announced Exper-Logo-Talk, a program written in ExperLogo which adds *speech* to the Logo capabilities of the Macintosh. It requires 512K and an external drive, and sells for $49.95. For more information, write to ExperTelligence, 559 San Ysidro Road, Santa Barbara, CA 93108, or call (800) 969-7874.

Logo users in central New York state are invited to join ASTROLUG, the Albany - Schenectady - TROy Logo Users' Group. For more information, write to Peter Rawitsch, PO Box 254, Guilderland Center, NY 12085, or call (518) 861-6637.

BIKILOG is short for Big Kids Logo Group, an organization formed to give Australian Logo users of all ages an opportunity to meet and interact. For more information, write BIKILOG, Box 88, Balaclava 3183, Australia.

Chris Templar would like information from anyone on techniques to move Commodore Microillustrator files into Commodore Logo. If you know a method to do this, please correspond with Chris Templar, Department of Computer Education, Johnson Bible College, Knoxville, TN 37998.

A free author's guide to writing articles for the NLX is available. To obtain your copy, send a self addressed stamped envelope to NLX Author's Guide, PO Box 5341, Charlottesville, VA 22905.

Learning Ukranian with Logo is a software development in Sprite Logo by Ihor Charischak. If you are interested in participating or would like more information, send for a free brochure. Ihor Charischak, Logo Computer Systems, Inc., 555 West 57th Street, Suite 1236, New York, NY 10019. (212) 765-4780.

Results are entered in the table shown below. (Be sure to tell your students about the control key in your version of Logo that retypes the line, so they don't have to type the whole line each time.) This phase of the experiment is the data collection step.

	Letters		Valid Combination?
S	PLOSIVE	LIQUID	
S	T	R	Yes No
____	____	____	Yes No
____	____	____	Yes No
____	____	____	Yes No
____	____	____	Yes No

Number of "yes" trials: ______

Total number of trials: ______

Percentage of valid combinations ______%

After the data is collected, students can begin the data analysis part of the experiment. Students use their knowledge of language to decide whether each combination of sounds is possible at the beginning of English words (the emphasis is on sounds rather than letters.) If there is dissention, you might have the class vote. Then the percentage of valid combinations is totaled.

It turns out that S and a PLOSIVE work well together. In many cases, the sound S, a PLOSIVE and a LIQUID can be combined to form a three-sound combination. In fact, about 50 percent of the combinations work.

	Letters		Valid Combination?
S	PLOSIVE	PLOSIVE	
S	____	____	Yes No
____	____	____	Yes No
____	____	____	Yes No
____	____	____	Yes No
____	____	____	Yes No

Number of "yes" trials: ______

Total number of trials: ______

Percentage of valid combinations ______%

In contrast, PLOSIVEs do not get along with one another. (That is, they don't make good blends.) The second data sheet shows this clearly. The percentage of valid combinations will probably be zero for this work sheet. The following commands are used to explore this combination.

```
?PRINT ( WORD "S PLOSIVE PLOSIVE )
SKG
```

Once you give students a model or a pattern, they can propose their own experiments. For example, which works better: combinations of FRICATIVEs and NASALs, or combinations of PLOSIVEs and NASALs?

Free-form experimentation can also produce unexpected results. For example, what might happen if we combine an S with a nasal sound and a vowel? If we knew the answer ahead of time, it wouldn't be necessary to do the experiment. Here's what might happen if we tried this particular experiment

```
?REPEAT 5 [ PRINT ( WORD    "S    NASAL
       VOWEL ) ]
SMO
SNE
SNGA
SMA
SNO
```

The combinations SNE and SMA are valid in English. (For example, they can be used to form "sneeze" and "small".) However, SNGA can never be used to form a valid English word. This approach might lead students to conclude that "ng" is never used in initial blends in English.

This way of using Logo for language exploration involves generating lots of examples and appealing to linguistic intuitions to decide which are valid. This approach has two things to recommend it.

- First, it makes the students more aware of the unconscious knowledge of English that they possess. It isn't necessary to memorize everything, because they already possess some of the information simply by virtue of being an English speaker.
- The second thing that is good about this approach is that the students are doing linguistics experiments in the same way that linguists do linguistic experiments. Papert has suggested on a number of occasions that children should do real science, not watered-down substitutes.

Another approach to language exploration relies less on intuition and more on pattern recognition. Using this method, students could look at examples of real English words and try to spot patterns.

Listful Thinking continued

Sounds Like

Thinking about how words *sound* compared to how they *look* provides an opportunity to confirm with students some of the regular irregularities of English spelling. For example, the letters we call "borrowers" are interesting under a microscope. The most obvious are C, X, and Q. They have no "sound" of their own, but simply borrow the sounds usually spelled by other letters. The letter C, for example, borrows the /s/ sound sometimes and the /k/ sound sometimes, depending on the rest of the sounds in the word. The experimenters should think of (or look up) a sample word beginning with each combination.

SC	A	SCAT
SC	E	SCENE
SC	I	SCIENCE
SC	O	SCOPE
SC	U	SCUM

What happens when they get to SCE and SCI? (The sneaky "C" borrows the /s/ sound: no more consonant blend!) When this strategy is used, the teacher has responsibility of salting the list of words with sufficient examples of the concept to permit the student to extrapolate the rule. (In same fashion, a science teacher does not send students out on the playground to collect random samples of elements. Instead, the combinations of chemicals provided for study are carefully chosen beforehand.) Some spellings texts have lists of words which are already organized in this fashion.

Other languages have a different chemistry of sounds. For example, Zbigniew Brzezinski presents an impossible combination of sounds in English, but not in Polish. In English, the combination ZB can never begin a word. In Polish, it can. For a look at the chemistry of the Hawaiian language, see last month's column.

A Rose by Any Other Name

In English, particular combinations evoke certain images. For example, why do *snert* and *smoot* sound funny? If you add a *k*, the combination is even better. For example, *Snookums* can bring a smile. Why does the word *Smurf* appeal to young children?

Another project we could attempt in our linguistic laboratory is creating a Logo program to produce names for a story. If the story is humorous, you might want a program that produces funny names, like the names above. Or you might want to create a program to produce science fiction names, for a story about monsters.

```
TO FUN.NAME
OP ( WORD "S NASAL VOWEL VOWEL PLOSIVE )
END
```

The program might work like this:

```
?PRINT FUN.NAME
SMEOK
```

Of course, the class will have to sift through the examples generated, to decide which ones are usable.

Name generators such as FUN.NAME provide a good entry point for discussions of word choice in creative writing. For example, Tolkien is an acknowledged master of style, and much of the appeal of *The Hobbit* lies in word choice. The reader knows immediately that a creature with the name *Gollum* must be up to no good. Ursula LeGuin's award-winning Earthsea trilogy is another example of children's literature in which not only do the names add richness, but also become an important theme. For younger children the work of Dr. Seuss is a perfect example, with *Grinches* and *Sneeches* abounding.

Components of a Logo Laboratory

A chemistry laboratory has equipment, raw materials, experiments, and lab projects. The laboratory allows students to experiment with concepts that they only read about otherwise.

In our Logo Language Lab, the raw materials are the categories of sounds that can be mixed together (PLOSIVE, NASAL, etc). The equipment consists of Logo procedures such as FUN.NAME which use these sound categories. The experiments described above suggest projects that students can conduct using the materials and equipment found in the lab. These experiments provided by the teacher serve as models for students, who may then create their own projects.

The lab and experiments described in this column have required a minimum of programming on the part of teacher or student. Instead, the emphasis is on language arts content. Discussing the sound system of English in a large group of students can be difficult if there is no way for students to gain experience with the concepts at some point in the lesson. A set of Logo tools makes such exploration possible for students working individually or in teams.

One feature which makes such a lab personally appealing to us is the likelihood of unexpected results. Chemistry labs have liquids and explosives, and our Logo Language Lab has LIQUIDs and PLOSIVEs. The excitement of true experimentation belongs not only in science, but also in reading and spelling instruction.

Listful Thinking continued

Mixing Agents for Word Formation

```
TO PLOSIVE
OP PICK [ P T K B D G ]
END

TO FRICATIVE
OP PICK [ S Z F V TH ]
END

TO LIQUID
OP PICK [ L R W ]
END

TO NASAL
OP PICK [ M N NG ]
END

TO VOWEL
OP PICK [ A E I O U ]
END
```

Logo Tools

```
TO PICK :LIST
OP ITEM (1 + RANDOM COUNT :LIST) :LIST
END
```

In Version 1 of Terrapin Logo, PICK requires the following procedures:

```
TO COUNT :LIST
OP COUNT.LOOP :LIST 0
END

TO COUNT.LOOP :LIST :NUMBER
IF :LIST = [ ] OP :NUMBER
OP COUNT.LOOP BF :LIST :NUMBER + 1
END

TO ITEM :NUMBER :LIST
OP ITEM.LOOP :NUMBER :LIST 1
END

TO ITEM.LOOP :NUMBER :LIST :COUNT
IF :LIST = [ ] OP [ ]
IF :NUMBER = :COUNT OP FIRST :LIST
OP ITEM.LOOP :NUMBER BF :LIST :COUNT + 1
END
```

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a reformed English major who studied linguistics at the University of Cambridge. She is now a speech-language pathologist working with language-disabled children.

TurtleTips

by
Jane Toth and
Donna Lanyi

In Search of the Disappearing Letter

As we've talked about in a previous column, many students often have difficulties with their typing skills. Some school districts are offering typing courses for elementary age students, while others believe that it is too early to concentrate on structured keyboarding. Whichever philosophy you believe in, it is helpful to have students become familiar with the keyboard to lessen their frustration and increase speed when working with Logo.

The Turtle Teacher

In our November 1984 column, we mentioned utilizing a postersize keyboard, desktop keyboards for students, keyboarding practice books, and a typewriter to strengthen keyboarding skills. We've expanded on these ideas through our own classrooms and with ideas from other teachers.

We enlarged the postersize keyboard into a trifold that is 5 feet long and 3 feet high. The keys are written on separate cards which are attached with velcro. Students love "ripping off" the keys and making a game out of who can replace the most keys correctly or the quickest. Normally we ask the students to concentrate on certain keys for a given period of time - home row keys, primitive command keys, special CONTROL keys, etc. We've seen this same idea done bulletin board size using styrofoam hamburger boxes for each key. This is a great manipulative activity.

Lucky us! Our PTO painted a giant keyboard on the playground! We use it for class activities, and the children love to jump out Logo commands during recess!

A bingo type game where students mark off the letters on individual keyboards is an old idea rejuvenated. We also supply blank dittoed keyboards for students to practice filling in.

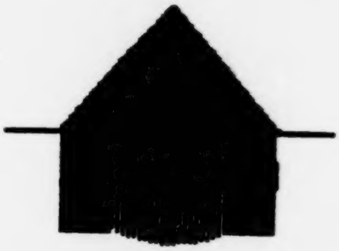
One teacher invented "Swat a Key." Primary students were divided into two teams and each team was given a fly swatter. As a letter was randomly called out, the team member to swat the key on an enlarged keyboard first scored a point for his or her team.

TurtleTips continued

Don't forget commercial software that develops typing skills. Students always love to practice a concept at the computer. Also, programs such as *MasterType* (Lightning Software, Inc.) allow you to set up a list of your own words for student practice. Special Logo primitive sets or groups of longer command names might be good candidates.

Procedure of the Month

Logo is bewitching! Have students design Halloween witches, masks, or creatures in Logo for bulletin boards, hall displays, party favors, or maybe even their own costumes! If you are interested in a listing of the procedures for the drawing below, please send us a self-addressed stamped envelope.



We invite any teachers or elementary students to send Logo procedures or ideas to be shared in this NLX column. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

Logo Notes

Kaleidoscopes, an outstanding Logo newsletter for experienced users, made its debut in May. Its editorial staff includes Alison Birch (co-founder and former director of The Phoenix School), Larry Davidson (author of the Apple Logo Reference Manual), and Phil Lewis (mathematics teacher at Lincoln-Sudbury High School) *Kaliedoscopes* is published bimonthly, and includes articles on applications of Logo at the intermediate and advanced levels, ideas for using Logo in secondary classrooms, and regular columns covering topics of interest to all Logo users. For more information, write to Kaleidoscopes, Computers for a New Education, 57 Waverly Street, Suite 100, Belmont, MA 02178.

Open-Apple is a monthly newsletter dedicated to providing full information about the ins and outs of the Apple family of computers to everyone. Each issue is packed with technical details and hints, tips on how to identify and fix bugs, and much more. If you are "into" Apple II computers, then you should subscribe to this outstanding newsletter. The editor is Tom Weishaar. To subscribe, send $24 to Open-Apple, 10026 Roe Avenue, Overland Park, KS 66207.

Back issues of *The National Logo Exchange* newsletter are available in sets. Volume 1 (September 1982 - May 1983), $15. Volume 2 (September 1983 - May 1984), $20. Volume 3 (September 1984 - May 1985), $25. Order from NLX, PO Box 5341, Charlottesville, Va 22905.

The NLX Research Directory is available for distribution. It contains the names and addresses of educators who are conducting Logo related research, and includes summaries of their work and interests. To obtain a copy, send a check for $3 to NLX Research Directory, PO Box 5341, Charlottesville, VA 22905.

Copies of the *Logo 85 Preproceedings* and *Logo 85 Theoretical Papers* are available for $15 per set. Send a check (payable to MIT Logo Group) to Logo 85 Preproceedings, MIT AMT, E15-309, 20 Ames Street, Cambridge, MA 02139.

Terrapin Inc. has Terrapin Logo version 3.0 available in packs of 10 ($299.95) and 20 ($525.00) language disks. An upgrade to version 3.0 is available for owners of version 2.0 for $20 per disk. For more information, contact Terrapin, Inc., 222 Third Street, Cambridge, MA 02142, or call (617) 492-8816.

Apple Computer, Inc., announces that both Apple Logo and Apple Logo II can be purchased in packs of 10 for the education price of $280. For more information, see your local Apple dealer.

Posy Lough's new book, *Christmas Decorations: Turning Family Treasures into Traditions*, contains instructions for making a wreath from Lego™ blocks. This decoration was inspired by the enthusiastic Logo activities of the children using Lego™ during the Logo 85 Conference. Other ideas from the book will be featured in the December issue of *Guideposts* magazine. To order a copy of the 40-page book, send a check for $5 (made out to The Posy Collection) to Christmas Decorations, PO Box 5341, Charlottesville, VA 22905.

Teacher to Teacher

A Review of New Logo Materials

by Anne Cairns Federlein

As we become entrenched in Logo this new school year, it is important to look at how Logo has affected the total school curriculum. Is Logo taught independently of other subjects? Has it has an effect on the teaching of other academic subjects in your school?

As students are encouraged to learn by discovery in Logo and to learn from mistakes in problem solving, a new way of thinking evolves for many children. It is my hope that educators will see changes in the children and convince administrators to allow the Logo philosophy to become an integral part of the total curriculum.

The following new texts have been reviewed for your classroom.

Beginning Students

Logo for the Apple II/IIe/IIc, by Maryann Marrapodi, 1985, published by Random House, 201 East 50th Street, New York, NY 10022.

Target Audience: upper elementary children

Content: The workbook includes chapters on Meeting the Turtle, Designing with the Turtle, Teaching the Turtle, and Circles and Colors. These are followed by Variables, Using Recursion, Turtle Stars and Spinsquares, Recursion and Spirals, and Turtle Projects. Chapters on Logo Words and Numbers, Lists and Sentences, and Using Conditionals round out the offering.

Strengths: The sequential format begins with introducing upper elementary children to the turtle. This workbook gives useful questions to ask the children during each chapter. It also allows the teacher to present information on Logo in the classrooom without every child using a computer. However, this method of teaching can be dangerous! Children learn by doing -- not by answering abstract questions in a workbook. Sufficient classroom time must be allocated for children in small groups, or individually, to play on the computer.

Teachers of early elementary children should use the book as a teacher's edition and simplify the instructions and questions. Older children will find the book stimulating, and the examples challenging. Teachers, even the novice, will find the book easy to understand.

Older Beginning Students

Apple Logo for Kids, by David A. Yule, 1984, published by Tab Books, PO Box 40, Blue Ridge Summit, PA 17214.

Target Audience: middle and high school students, Logo teachers, and parents

Content: The turtle is named Sam and appears in chapters such as: An Introduction to Parents, Teaching Sam How to Do Things, What to Do If You Are Stuck, If You Don't Type Well, You Will Love This!, Using Color with Sam, and Telling Sam to Tell Sam to Tell Sam ...

Strengths: This book makes learning Logo fun for both the student and teacher. The immediate technical and curricular concerns of the beginning student and teacher are addressed in a playful manner by naming the turtle Sam, and by giving the chapters titles that capture the imagination. (Can you guess what the "You Will Love This!" chapter is about?)

The format of the text does not have the workbook appearance of many beginning Logo books. The written text is high level and the authors have discussed all the practical concerns of beginners in easy-to-understand language. Each chapter contains a drawing of a watchdog marking important information to remember.

Questions and crossword puzzles give the student and teacher an opportunity to review the information presented in the chapter. These reinforcement activities are typical of the ones used in schools today and, all too often, replace class discussion. Teachers should remember the value of children learning through their mistakes, rather than recording pat answers to the rote questions. We run the risk of making Logo just another school subject with drill and practice activities for learning.

The appendices contain a useful parent's guide and information on how to get started with both Apple and Terrapin Logo, along with equivalent commands and keystrokes to "translate" from Apple Logo to Terrapin Logo, TI Logo, and TRS-80 Color Logo. This text is very useful for introducing older students, teachers, and parents to Logo.

Do you know of any new Logo materials and books? Did you find them useful in teaching? If so, we invite you to send information to The Logo Information for Teachers (LIFT) Group, PO Box 5396, Plymouth, MI 48170.

Anne Cairns Federlein is a professor in the Early Childhood Area, School of Human and Educational Services, Oakland University, Rochester, MI.

Q and A

by
Jim McCauley

Q. I'm used to programming in BASIC, and though I think Logo is a great language, I really miss having arrays to work with. Why doesn't Logo have arrays?

A. Arrays closely match the BASIC and FORTRAN models of what a computer is: a register machine. That is, data of highly restricted types in highly restricted formats are stored in address locations which (usually) are contiguous in the machine. For example, if you want to store a list of 16-bit signed numbers in a register machine, a very natural way to do it is to store them in immediately adjacent pairs of bytes in memory. They can be accessed very conveniently (and quickly) by means of an index; in BASIC, one would recall the third element in an array called "ARRAY" by calling ARRAY (3).

Conveniently, most array-processing languages (such as BASIC, FORTRAN, and Pascal) make provisions for accessing elements of multi-dimensional arrays, thus greatly simplifying the problem of constructing tables. For example, one could reference the third element of the second "list" in a two-dimensional array called "ARRAY2D" by referencing ARRAY2D (2,3).

Array with Logo

In Logo, though, information is stored in linked lists, and construction of an index is (usually) difficult or impossible. Instead of accessing items in lists by reference to indices, the usual method in list-processing is to decompose a copy of the list recursively until the item desired is reached. Arrays may be simulated through the use of the ITEM primitive, and multi-dimensional arrays may be constructed from "lists of lists."

Changing contents of such a simulated array, however, will always require the use of recursion. This might sound terribly inconvenient, but it actually makes a lot of sense. In an array processor, you must concentrate on manipulating pointers (indices), but in a recursive list processor, you are manipulating the data *directly*.

Thus, a level of potentially confusing indirections is eliminated.

Up, Up and Array

There are two versions of Logo in which arrays have been implemented: *ExperLogo*, a compiler from ExperTelligence (559 San Ysidro Road, Santa Barbara, CA 93108. See the May 1985 NLX for a brief review.), and *[TLC] Logo*, from The Lisp Company (430 Monterey, #4, Los Gatos, CA 95030. Also see *Thinking about [TLC] Logo*, reviewed in the March 1984 NLX.) [TLC] Logo implements arrays as vectors, a very sophisticated technique borrowed from another computer language called APL.

I hope this background information gives you encouragement to "play around" with lists of lists and recursion in Logo.

Each month, I will answer selected questions in this column about Logo programming techniques or the thinking which leads to programming. If you have a question, send it to: Jim McCauley, 876 East 12th Avenue, Eugene, OR 97401. Please enclose a self-addressed stamped envelope.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

LOGO NOTES

The first *World Logo Conference* is scheduled for October 25 - 27, 1985, and is an event you shouldn't miss! Offered through CompuServe, the conference makes it possible for you to participate without leaving home. Online panel discussions with well known Logo workers will be featured, in addition to other activities. Panelists include Seymour Papert, Molly and Dan Watt, David Thornburg, Brian Silverman, Jim Muller, Judith Pope, Rina Cohen, Dale Burnett, Bill Higginson, Uri Leron, Paul Goldenberg, Tom Lough, and many others. Participants from many countries around the world are expected to log in during this electronic festival.

If you are already a CompuServe member, registration is $15. Otherwise, the cost is $55, and includes a CompuServe membership. It is necessary to use a modem to participate in the conference (and other CompuServe activities). For more information, write to World Logo Conference, 1616 Charles Street, Vancouver, BC Canada V5L 2T3.

Are you looking for a new dimension to your teaching this year? The *Logo Class Penpal Network* may be just the thing! By sending in an application, you and your class will be matched with a counterpart from the USA, Canada, or perhaps one of several overseas countries, based on grade level, computer compatibility, Logo experience, and interests. During the school year, students and teachers exchange Logo ideas, procedures, and projects. For a free application, send a self addressed stamped envelope to Logo Penpals, School of Education, University of Virginia, 405 Emmett Street, Charlottesville, VA 22903-2495.

NLXual Challenges

by Robs Muir

The BASIC Challenge

This NLXual Challenge is unique in several ways. First, unlike all previous Challenges, I know currently of no existent solution. (Have I sparked your interest?) Second, this Challenge presents a potentially more complex and lengthy project than those during the past two years. Thirdly, this project's solution may present the most compelling answer yet to my students' questions about when "...will we begin to learn BASIC?" So, it is with a mixture of computational revulsion and perverse curiosity that I present this month's Challenge.

Back to BASIC?

I think the most difficult Logo Challenge I've faced as a teacher has been the one posed by several of my students at least once every year. Perhaps you've experienced it in your Logo classroom. The scenario runs something like this.

After months of careful planning, you've finally managed to create the perfect Logo environment, one rich in student-initiated learning activities that integrate your curricula into your students' lives and, incidentally, one that makes much use of the computer (or computers) in the back of your classroom. Every kid is having a ball and your level or creativity has never been higher.

Then dawns that fateful day when the bubble bursts and the spectre of "computer literacy" rears its ugly head ... the day when your "hottest" 10-year-old Logo programmer asks,"Teacher, when are we going to learn BASIC?"

Of course, you try to explain Dijkstra's warnings about GOTOs, the power of procedural / functional description, recursion, lists as data structures, Lambda calculus, etc., but you begin to sense that your remarks are falling on deaf ears. After all, your young inquirer has read some books on computers (all written for BASIC users), and his or her older friends all know BASIC. They have computers at home (which have only BASIC, because who can afford a Logo disk), and besides, they're taking "computer" in high school (where they teach only BASIC!).

Can you blame your student for wanting to "graduate" to a perceived higher level of programming? The "samba school" of the street is overwhelmingly in support of line numbers.

Where is the really convincing demonstration of Logo's expressive power that a fifth grader can understand? I've been searching for such a demonstration since I began to notice this adolescent preoccupation with QWERTY languages to the exclusion of better tools like Logo. The solution to this month's challenge may well be it!

Brian's Sparks

Brian Harvey's new book, *Computer Science Logo Style*,(MIT Press, 1985) contributed two sparks that ignited this Challenge.

* Spark number one: As a pedagogic trick, Brian suggests a Logo demonstration during that first lecture. In the first few weeks, new Logo students can't be too inventive with a new language; "That's the one reason I like to start by showing off a more complex project the first day. It helps me convince the students to have faith in me for the first week or two while they're limited to trivial projects themselves."

* The second spark is found in Brian's "sneaky greeting."

```
TO START
CLEARTEXT
PRINT [ WELCOME TO LOGO. ]
TYPE "?
PR SE [ I DON'T KNOW HOW TO ] FIRST SE RL
TYPE "?
WAIT 100
PRINT [ HA, HA, FOOLED YOU!! ]
END
```

Do you see the challenge yet? What better way to show the power of Logo to an incorrigible BASIC fan than to write a complete BASIC interpreter in Logo! Can we do it? Intuitively, I'd say, "SURE!" But I'm still working on it.

First, we've got to get rid of those overly informative Logo error messages; how unBASIClike! If your Logo implements CATCH and THROW (look these up in your Reference Manual), you might try this:

```
TO BASIC
CATCH "ERROR [ RUN.COMMAND ]
PR [ SYNTAX ERROR ]
BASIC
END
```

Challenge continued

```
TO RUN.COMMAND
TYPE CHAR 93
RUN READLIST
RUN.COMMAND
END
```

This program, BASIC, will run Logo just fine, except the old Logo prompt (the "?") will be replaced with a "[". However, we will always get a rude SYNTAX ERROR statement if we type something like FORWORD 100. (By the way, a CONTROL G won't abort this abomination of a program; use THROW "TOPLEVEL.)

Interpreting BASIC

I suppose the ultimate challenge here is to write a complete BASIC interpreter in Logo that would run any standard Microsoft BASIC program. There are a host of related complexities you'll discover as you tackle this project. It may not be possible on an Apple, a TI, or an Atari, due to memory space. But imagine the possibilities with an IBM PC, an Atari 520ST, a Macintosh, or an Amiga.

On January 1, 1986, The National Logo Exchange will award a free copy of Brian Harvey's new book to the most complete (e.g., best Sieve of Erathosthenes benchmark) BASIC/Logo interpreter. Reprints of all past NLXual Challenges will be awarded to the most complete 6502 implementation. So much for intrinsic motivation!

Send your completed program listing, with a stamped self-addressed envelope to

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, CA 91711

We will send copies of the winner's solutions to all contributors. In addition, if you would like to suggest an NLXual Challenge for other NLX readers, send your problem to the above address.

```
10 PRINT "FORWARD 100"
20 END
```

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.

Imagewriter Overprinting Fixed!

Thanks to Mark Eckenwiler of Terrapin, Inc., for furnishing this Terrapin Logo procedure which prevents the Apple Imagewriter from overprinting long lines in Logo procedures. Type in the SETUP procedure, turn your printer on, run the SETUP procedure, and then print out your other procedures in the usual way. (Ed. note: See "Imagewriter Overprinting?" in the September 1985 NLX.)

```
TO SETUP
OUTDEV 1
(PRINT1 CHAR 9 "C CHAR 13 )
OUTDEV 0
END
```

Steve Tipps wrote and tested this corresponding procedure for Apple Logo.

```
TO SETUP
.PRINTER 1
(TYPE CHAR 9 "C CHAR 13)
.PRINTER 0
END
```

Logo Note

The *West Coast Logo Conference* is being held November 21 - 23, 1985, at the Pacifica Hotel in Culver City, CA. Many panels, discussion sessions, workshops, and presentations are planned to make this one of the major Logo events of the year. Send now for your registration materials. West Coast Logo Conference, Pepperdine University, 3415 Sepulveda Boulevard, Los Angeles, CA 90034.

The National Logo Exchange, copyright © 1985 by Posy Publications, a part of The Posy Collection, all rights reserved.

Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717.

Editor..Tom Lough

THE NATIONAL
LOGO
EXCHANGE

The Logo Newsletter for Teachers

Volume 4 Number 3 FORWARD 100! November 1985 Page 1

Teachable Logo Moments: Jimmy Comes to Class

by Louise Robinson

"Jimmy, how did you do that?" asked the 22-year-old graduate student.

Jimmy, 6 years old, had accompanied his mom to her evening graduate class, Logo in the Curriculum, and was printing his Logo project on the printer.

Jimmy's mom had approached me earlier about the possibility of bringing him to class occasionally. Parameters were set and permission granted. As the Logo "coach," I had no idea such a valuable teaching aid was being added to my course!

Good Evening, Class

The evening started off quietly enough. Jimmy played TURTLE.TARGET at the computer (his mom had done some pre-computer activities with him before coming to class) while we discussed the first few chapters of *Mindstorms*. He listened as I gave the evening's challenge to the graduate students: to create, save, and print a picture. The students and Jimmy went right to work.

The graduate students, with all the courage they could muster, drew with preconceived plans and chose "conservative" commands such as FD 10 and FD 30 ... everyone used numbers 50 or less.

Not Jimmy! His drawing, without a plan, went like this: FD 100 RT 23 FD 1000 LT 36 FD 10000 etc., until he had almost filled the screen.

The graduate students helped each other print out their first "masterpieces." They commented on how quiet the printer was ... mostly just humming, with an occasional staccato of printing noise. I explained to them that the noise level was related to what was being printed; most of their drawings had just a few relatively short lines.

How Did You Do That?

It was Jimmy's turn to print out his picture. The printer broke into a cacophony of activity! That's when the graduate students turned to him in amazement and asked, "Jimmy, how did you do that?" Jimmy replied, "I'm not sure, but I just kept moving the turtle, trying different numbers."

Always being one to capitalize on the "teachable moment," I concluded the class meeting with a discussion about how children are more willling to take risks that adults. I could have tried to convince the graduate students of this fact, as a result of my Logo teaching experience with young children. However, I didn't have to; Jimmy had done it for me! (There's nothing like being in the right place at the right time. I hope Jimmy will be able to come again!)

Let's Get Serious

Most undergraduate and graduate college students who take Logo courses hope to teach Logo to children and to other teachers. Those of us who teach such courses need to give serious consideration to how we approach the class.

Teachable Moments continued

My experience is that most college students have little difficulty learning Logo primitives and procedure writing. The real challenge lies in communicating the Logo philosophy and providing them the opportunity to show evidence of understanding and to apply it within the curriculum. My contention is that this is the most difficult (and the most important) task for the professor.

In my courses, I try to accomplish this is a variety of ways. The most obvious one is creating a Logo environment in the college class. Here are some ideas.

1. Allow students to choose their partners.
2. Implement and stress well-known Logo sayings, such as:
    "Try it!"
    "Ask three before me."
    "Tape it, walk it, turtle talk it."
3. Combine direct instruction with individual coaching.
4. Encourage student input for all decisions on assignments and evaluation procedures.
5. Stress that errors are not mistakes, but simply new problems to solve, then demonstrate that you mean it!
6. Provide a variety of Logo resources in class for students to consult if they "get stuck."
7. Consider not giving quizzes or objective type tests. Start each class wih a review of the class work from the previous session. Use a variety of evaluation techniques weekly.
8. Allow each student to create a personal Logo project during the semester and to share it with the rest of the class at the end of the course.

I'm constantly reflecting on ways to improve my college level Logo courses, and adding new ideas to the above list. Jimmy has caused me to consider seriously the idea of inviting children with no previous Logo experience to attend my classes.

Thanks, Jimmy! Whoever said, "Learning shall flow from the older to the younger," never met you!

(Ed note: Have you experienced a "teachable Logo moment" in which the lesson plans went out the window and spontaneity reigned? We invite you to share these special moments with other NLX readers. Send your notes to Teachable Logo Moments, NLX, PO Box 5341, Charlottesville, VA 22905.)

Louise Robinson is a professor at Indiana University of Pennsylvania, where she teaches graduate and undergraduate Logo courses. She is also a third grade teacher in the university laboratory school.

From The Editor

by Tom Lough

Guest Editorials: Logo Teacher Training

Teacher training is critical to the success of teaching with Logo. Sensitive Logo trainers such as Louise Robinson provide examples of successful experiences. But are these experiences the rule or the exception? I have asked two other teacher trainers to give their perspectives and suggestions in this month's editorial column. I encourage you to correspond with them and share your viewpoints and responses.

FD 100!

Tom

Elaine Biltman:

"I just got assigned as my school's computer specialist, and I don't know a thing about computers!"

"The principal told me I'd better learn Logo over the summer because our school is getting ten computers in the fall, and Logo will be part of the curriculum!"

Most conscientious teachers spend large amounts of "leisure time" and their own funds to keep abreast of new research, technology, content, and methods. Some school districts encourage their teachers by giving in-service courses, stipends, and salary benefits to those who keep up-to-date.

Other systems, however, mandate changes in curriculum or assignment, expecting the staff to become competent on their own initiative. This sort of treatment tends to set up ambivilant feelings in the most enthusiastic of educators, but most of them dutifully meet expectations (take the course, set up a Logo program, etc.) to the best of their ability.

Quite often, however, what classroom teachers get for their extraordinary effort is a variation of an all too prevalent negative theme: *Teachers don't know how to teach and now they can't teach Logo, either.* This theme has even been expressed in workshops and discussions at national computer education conferences, when one would think that professional encouragement and appreciation for classroom teachers should be expressed.

Not only is this type of generalization unfair, it's untrue! There are excellent teachers already using innovative techniques for helping children learn to think, who are now helping them learn to think with Logo. If Logo isn't being taught as well as it should be in some

Guest Editorials continued

instances, I feel the teacher-training courses (and especially the follow-up support) should be assessed, not the teachers!

If Logo is to create a revolution in education, teachers who are learning it must be taught with the spirit and philosophy of discovery, exploration, and problem-solving, and above all with a positive attitude by their own instructors toward their students' accomplishments. Given the structure of the usual university or district in-service course, with requirements for a certain number of credit hours, papers, evidence of competence, or "covering" a certain amount of course material, this is difficult, but not impossible.

A creative instructor can make requirements fit the needs of individual teacher-students, and also instill the desire to continue discovering ways to teach and learn Logo with one's own pupils when the course is over. Follow-up opportunities with an instructor and/or colleagues must be available for teachers after the course is over.

Most courses provide sufficient (and sometimes an overwhelming amount of) information, including bibliographies of materials, books, and newsletters. All of these are helpful, but teaching is often a lonely business.

What teachers really need after their training and during classroom experience with children and Logo is constructive encouragement: exchange of ideas about methods that work and those that don't; classroom management hints; technical problem solutions; insight into children's reactions; reinforcement of the Logo philosophy; and the knowledge that others are interested and eager to assist in creating a successful Logo program in that individual's classroom.

Research tells us that we should reinforce the positive aspects of those behaviors we want to nurture in children; *surely* those same principles apply to computer education for classroom teachers!

Mary Lou Miller:

Logo teachers are often products of their training ... some are good, some could care less. But, with appropriate organizing within the school, the needs of the students can be served.

When I first began working with Logo four years ago, Abelson's book ws the only resource available. At that time, a few teachers were using Logo with their students and showing their peers the fantastic capabilities of the language. As the popularity increased, the amount of materials grew so that today there is such an abundance of materials it is difficult to make appropriate selections.

With the increase in popularity of the language, school districts began to include Logo as part of the computer curriculum. With Logo now being taught in many schools, I am seeing different types of Logo teachers emerging. One group consists of those enthusiastic teachers who are trying to create the "microworlds" for students.

Another group consists of those who have been given a "crash course" in Logo (e.g., a three hour workshop), handed a curriculum guide, and then told to integrate Logo into the curriculum. These teachers have such minimal knowledge of the language and such insufficient training that they either postpone teaching it until the last minute at the end of the year, set aside just one or two weeks where they will teach Logo once a day for that time period, or merely forget about it altogether.

From my perspective, not everyone is meant to be a Logo teacher. An enthusiastic teacher well versed in the language who believes in the Logo philosophy could provide the primary instruction in Logo to the students, with the classrom or content teachers designing the follow-up activities. Organizational teaching techniques such as this help can help students go beyond the syntax and use the language as a tool to develop thinking and problem-solving skills.

Assisted by the guiding hands of knowledeable instructors, I feel very strongly that Logo will continue to thrive in our schools and be one of the many tools available to assist us with the important task of teaching problem-solving skills to our students.

Elaine Blitman is the K-2 supervisor at Punahou School, 1601 Punahou Street, Honolulu, HI 96822. She has taught a number of Logo courses for teachers and is the co-author of a series of Logo charts and cards published by Scott, Foresman and Co.

Mary Lou Miller is an instructional computing teacher in nine school districts for the Monroe-Orleans BOCES, 3599 Big Ridge Road, Spencerport, NY 14559. She also teaches summer computer courses at nearby Nazareth College.

Logo Note

Logo in the Schools, a special double issue of the journal *Computers in the Schools*, contains 30 articles by Logo workers of many different orientations. This highly recommended volume of over 300 pages is available in softcover for $7.95, with a minimum order of 5 or more copies. If you are involved with Logo teacher training, this is an excellent resource to consider. For more information, contact Haworth Press, Inc., 28 East 22nd Street, New York, NY 10010, or call (212) 228-2800.

Tipps for Teachers

by Steve Tipps

Community Builders

Take a walk around the school with your students. What do you see? Buildings and trees and houses and parks and roads are likely sights. When you come back to the classroom, children bring with them experiences which can enrich language, math, science, and social studies.

Follow-up with a map making project. Map making incorporates skills from all the curriculum areas. The topic in last month's column was making classroom maps with Logo. This month, Logo projects are suggested for constructing neighborhood or community maps.

Turtle Maps

An early map activity included in many instructional books for Logo is a map drawn on an acetate sheet. The sheet is taped over the screen. The turtle is then moved around the screen from the grocery store to the pizza place to the laudromat and back. Students enjoy this activity as they begin to move the turtle.

Turtling with the map can be done with regular distance and direction commands or a **DRIVE** procedure, such as that found in *Learning with Logo* (Dan Watt, McGraw-Hill, 1983). A simpler procedure without recursion could also be written. Using this procedure as a template, children elaborate by adding more options.

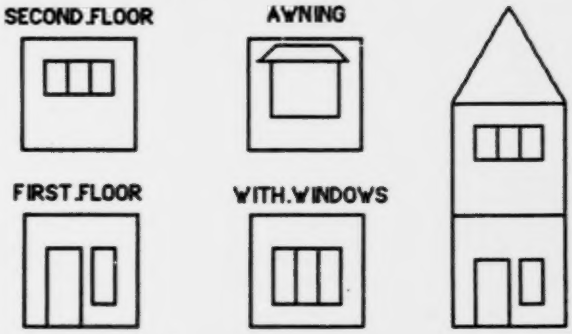
```
TO MOVE
MAKE "KEY READCHAR
IF :KEY = "G [ FORWARD 5 PRINT "GO ]
IF :KEY = ..................
IF :KEY = ..................
IF :KEY = ..................
END

REPEAT 100 [ MOVE ]
```

MOVE is powered by single keys which can be labelled with stick-on covers. The strengths and limitations of single key Logo procedures have been discussed previously in this column (See "The Issue of Instant" in the December 1983 NLX). I suggested that **R** and **L** are not very good keys for **RIGHT**and **LEFT**. **R** is on the left side of the key board, and **L** is on the right. The kinesthetic learning for **L** and **R** is wrong. The concepts of clockwise and counter-clockwise may be better ideas to emphasize.

The map you use can be the actual neighborhood of the school. Draw a simple road map on acetate and invite the children to add things to the map with washable projection markers. The same acetate can also be used on the overhead projector for class activities in mapping.

Houses

Houses are one of the first projects which many people tackle, whether they are 4 or 64. A square house with a triangle roof almost always appears at an early stage in Logo learning. This protohouse can be elaborated in many ways. Mark Mariotti, of Jefferson County, Colorado, challenged his students to make other kinds of houses. When I visited his class, the problems ranged from getting the door just right to building a picket fence to putting curtains in the windows. These houses were glorious examples of Logo at work. Each team of programmers had their own goals and their own approach to house building.



Mark had also helped the children increase their awareness of their environment. Denver is full of grand, old homes with grillwork and gargoyles. He had photographs and diagrams of architectural details on the

Tipps continued

bulletin board as beautiful reminders of new challenges for Logo. Almost every community has some area which is preserved or restored or which should be. Look around for houses in your community which should be cherished.

Public Places

Another source of inspiration would be the schools, hospitals, churches, libraries, and monuments in the community. Delores Lutz, of Charlottesville, Virginia, had the students in her fifth grade class create a travelogue of Washington DC. Programming teams selected the Washington Monument, Lincoln Memorial, Congress, or the White House to render with Logo. Text was added to describe what the children saw on a field trip.

A trivia quiz could also be made about the history of the community.

```
TO QUESTION.1
PRINT [ WHAT IS THE OLDEST BUILDING IN TOWN? ]
MAKE "ANSWER  READLIST
IF :ANSWER = [ POST OFFICE ] [CORRECT.1  THROW
      "TOPLEVEL ]
HINT.1
QUESTION.1
END

TO CORRECT.1
PRINT [ THE POST OFFICE BUILDING WAS BUILT IN
      1887. ]
PRINT [ THE LIBRARY IS NOW IN THE OLD POST
      OFFICE BUILDING. ]
PRINT [ IT WAS MOVED THERE IN 1983. ]
PRINT [ LOOK AT THE FRIEZE ON THE FRONT OF
      THE LIBRARY.]
PRINT [ DO YOU SEE ANY ONE YOU RECOGNIZE? ]
END

TO HINT.1
PRINT [ ONCE A PLACE OF LETTERS,  NOW  IT'S A
      PLACE OF BOOKS. ]
END
```

Simple quiz templates like this one should result in a huge list of trivia questions for parents at the next parent's visitation. (How would you randomly generate questions for parents? A hint is to make a procedure which calls a second procedure named with **WORD "QUESTION. RANDOM 50** if you have fifty questions.) Children soon find out that they can write questions on every topic.

Formulating and phrasing questions is an important study skill. The teacher's task is to encourage involvement of students rather than creating quizzes and tests for students.

Community Construction Sets

Another project is to create a community construction set of Logo procedures. A construction set contains pieces which can be used to build a bigger project. Sounds like Logo already, doesn't it?

Look at an Erector set with standard pieces which can be used to build many things. Lego™ pieces have the same modular nature. Jane Toth and Donna Lanyi report on a Logo project such as this in their *TurtleTips* column in this issue.

The most famous electronic construction set is Pin Ball Construction by Bill Budge. With this great program, you build your own Pin Ball machine. Rocky's Boots and Robot Odyssey, both from the Learning Company, have much the same structure.

The Logo community construction set would be made of procedures needed to draw your community. The list from the field trip could be used to start: house, apartment building, school, church, factory, filling station, etc. Natural features would also be needed: trees, parks, ponds. Procedures for each piece would be written so that **HOUSE** would draw a house and **CHURCH** would draw a church.

The acetate sheet could be used for laying out the roads. Instead of just driving the turtle around this time, the students would "build" buildings along the roads.

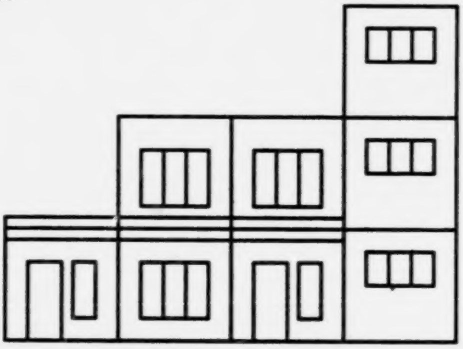
The procedures could be programmed ahead of time by the teacher. Janet Tevendale of Charlottesville made a community construction set like this for her third graders. Or, making the procedures could be a student project. I prefer the latter, but the way you handle this depends on the goals and the time devoted to it. You might provide only a few examples as motivation for their work.

After listing the pieces needed for the community, the class may need to decide on a scale for them. Houses, apartments, stores, and factories differ in size, but they are usually in scale. A basic unit might be a 20X20 turtlestep square if you are going to try to show several streets the screen. A construction set for a single street might utilize a 30x30 square or larger as the unit.

A house might be a simple protohouse with triangular roof. An apartment building might be several units stacked beside each other.and on top. A factory could fill a whole block. Decisions about these construction pieces and how they fit together are the important problems in this project. These decisions are

Tipps continued

exercises in modular thinking. Look for units in existing buildings. How are the windows arranged? Where are the doors? Does the same window pattern occur several times?



For children in the upper elementary grades, community construction is a good way to deal with issues of zoning and community planning. Why are stores built on main roads? Why are houses put together in a different part of town? What are the advantages of having people live downtown? Issues of climate and terrain are also important to consider.

A visitor from the city planning office would be a good resource after children have worked with various community arrangements. Charts and maps from the planning office provide a real life view of the process.

A community construction set was created for second graders at Thomas Jefferson Elementary in Falls Church, Virginia. Indian communities and a kibbutz were part of the social studies curriculum. The teachers offered Logo as another means of planning and constructing the communities.

Community Workers

One of the benefits of a project which takes many hands is that everybody learns from everybody else. Whether the learning is among the children in one class, shared between several classes, a combined fifth grade-second grade project, or a cooperative effort by teachers, the problem solving, arguing, and solutions are better than achieved by a single programmer.

The shared project is crucial in creating the kind of learning community which Papert envisions. Individual projects are still important, but isolation is not desirable. After all, Logo community builders are building two kinds of communities.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX. He conducts Logo workshops for school systems throughout the United States and Canada.

Logo Teacher Feature

by

Rebecca Poplin

Featuring: Sebastian and Sandra Reisch

Sebastian and Sandra Reisch make their home in rural Ontario, Canada. Their house is built around a 150-year-old farmhouse that they have remodeled and expanded. It sits atop a gentle hill overlooking the land that they farmed when their children were growing up. Both Sebastian and Sandra are now employed by the Northumberland/Newcastle Board of Education, which serves an area about an hour's drive from Toronto. Sandra teaches first grade in a special French immersion program. Sebastian teaches mathematics and computer science in high school.

For the Reisches, the long trek from Ontario to Charlottesville, Virginia, was just another step toward learning how Logo could help them in their classrooms. They came to the University of Virginia in the summer of 1984 to take a Logo course taught by Steve Tipps and Glen Bull.

Sebastian and Sandra came to Logo from different directions, and have continued with Logo at opposite ends of the educational spectrum.

Sebastian has been interested in computers from the very beginning; in 1969 he became one of the first teachers of computing in his area. He searched for anything and everything he could get his hands on to learn more about computing.

One of the discoveries Sebastian made was the work of Richard Pattis, a Stanford professor who developed Karel the Robot as a teaching tool for his Pascal students (Pattis, Richard E. *Karel the Robot*. New York: John Wiley & Sons, 1981). Karel's language is very Logo-like in that he is moved on the screeen on a system of streets and avenues with moving and turning primitives. In addition to turning and moving, Karel picks up "beepers" at different locations. Actually, Karel's language is a subset of Pascal, with Pascal structures but no variables.

Initially, Sebastian's students wrote programs and moved Karel around without a computer. Later on, using Logo, they implemented Karel's language on the computer. They wrote a translator, a compiler, an editor, and a linking program as a major group project for the class. Sebastian might participate in a research project with Rina Cohen at the Ontario Institute for Studies in Education in Toronto. They are interested in comparing two groups of college preparatory mathematics students, one group using Logo and the other using traditional methods.

Teacher Feature continued

Sandra has been a teacher of French for thirteen years. At first, she was an itinerant teacher, giving 20-minute daily lessons in grades 1-10. Eight years ago, she was selected to work in a French immersion program for first grade English-speaking students whose parents wanted them to become bilingual. She teaches all the customary first grade subjects, except that she teaches in a language which is not her students' native tongue.

Although she was aware of computers through Sebastian's interest, it was not until Logo became available that she felt computers had something to offer her. Two years ago, she began using a French version of Logo with a customized Instant program she wrote.

Sandra's version of Instant has several interesting features. She uses the French words for ball, box, and hat for circle, square, and triangle, and has developed an erase command which erases the previous line of the child's drawing.

Sandra maintains a sense of overall curriculum sequence in her teaching. For example, she tells her students that, when they go to second grade, they will be working with the turtle's cousin who doesn't know how to do ball, box, and hat, so they will have to teach the cousin these new words.

One of Sandra's many projects for the coming year is developing a data-base dictionary for her first graders.

A visit with Sebastian and Sandra Reisch provides a wonderful illustration of the diverse uses of Logo in schools. Although the contexts in which they use Logo are very different, Sandra and Sebastian have in common a great enthusiasm for what Logo can and has done for their students. They also share an intense desire to learn and grow and to share their ideas.

Sharing ideas is a big part of what the NLX does for teachers. If you have ideas you would like to share through this column, or if you know of teachers who should be featured, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, Texas 76308
(817) 766-5323

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

ExperTelligence announces *French ExperLogo* for the Macintosh. For more information on this French version of ExperLogo, write to ExperTelligence, 559 San Ysidro Road, Santa Barbara, CA 93108, or call (805) 969-7874.

Logo Disserts:

Dissertations Dealing with Logo

by Barbara Elias

Computer educators involved in Logo instruction for adult audiences may find the results of an observational study by S. C. Murphy of interest. Murphy investigated the planning, programming, and debugging strategies of 25 adults (20-year-old female undergraduate students, for the most part) , in a series of Logo programming sessions, *without* instructor assistance.

Following a tutorial session, subjects spent about five hours in programming sessions of 1 or 3 hours each. The tutorial, with assistance, was designed to acquaint subjects with Logo in the immediate mode, and to prepare them to use the edit mode to write a simple procedure as well as a procedure which used variables.

Programming sessions were scheduled for each subject, individually, with the researcher present only to observe and record behaviors. An Apple II+ microcomputer was used in developing chosen programs. Paper, pens, pencils, markers, a compass, and a ruler were available for use. No instructor assistance was available; however, three Logo textbooks, notes, and self-made materials could be used.

Each subject was observed for about five hours, or until an original recursive procedure was written. Behaviors were recorded in planning, programming, debugging, and testing categories. Exit interviews helped determine reasons for preferred behaviors.

Results indicate that planning, programming, and debugging strategies covered a range of behaviors. Planning strategies exhibited ranged from copying procedures from the text to using text materials as a resource. for creating original procedures. Debugging strategies were noted in three categories: building (which used primarily the immediate mode), adjusting (which used the edit mode), and typographical corrections. About one-third of the subjects used no debugging strategy. There was little change in strategies throughout the programming sessions.

Those who created new information were also in the group who used debugging. Two subjects created recursive procedures unassisted. All reported that learning on their own was a rewarding experience.

>>>Murphy, S. C. (1984.) *Adult Processing Characteristics When Using the Logo Computer Programming Language* (Doctoral dissertation, University of Texas at Austin, 1984). *Dissertation Abstracts International*, 44, 1977-A.

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of Virginia.

Listful Thinking

by Glen Bull and Paula Cochran

Words, Names, & Starmaps

In last month's editorial, Tom Lough discussed the importance of names. We're not quite ready to change the title of our column to "Nameful Thinking", as he suggested. However, we will take that as a cue for a column on names.

Colorful Names

Logo allows you to use names for numbers. Most versions allow you to change the color of the turtle's pen. The command below sets the turtle's pen color to blue in Apple Logo. (If you are using Terrapin Logo, substitute PC for SETPC.)

```
?SETPC 5
```

However, the number 5 represents a different color in Commodore Logo, while there is no pen color 5 at all in IBM Logo. There's no particular virtue in requiring students to remember that 5 means "blue" in one dialect of Logo. In fact, using the color words themselves might do more to complement academic goals. Here's a procedure that will allow us to use the word BLUE, instead of using the number 5.

```
TO BLUE
OUTPUT 5
END
```

Now when we say BLUE, the computer thinks "5"

```
?PRINT BLUE
5
```

This allows us to use the word BLUE instead of having to remember that 5 represents BLUE.

```
?SETPC BLUE
```

At this point, we have to confess. This column is really about the OUTPUT command. We just used OUTPUT to make the name BLUE. One of our brighter students objected to making BLUE in this way. Here's how she wanted to make BLUE. (We'll call it BLUE.2, to avoid confusion.)

```
TO BLUE.2
SETPC 5
END
```

She argued that this approach was more efficient, because you only have to type one word to change the pen color. She would be correct, if we were only going to use BLUE to change the pen color. However, our BLUE can also be used to set the background color as well. (If you are using Terrapin Logo, substitute BG for SETBG.)

```
?SETBG BLUE
```

Of course, you don't want to type in the names of the colors from scratch each time. Instead, create procedures for all the colors. Then save the procedures on your disk in a file called COLOR.NAMES.

```
?SAVE "COLOR.NAMES
```

Now whenever you or your students are working on a project that uses different colors, you can load in the COLOR.NAMES file and use the names of the colors, instead of their numbers. We suggest that teachers create a Logo Class Tools disk that contains files of useful procedures such as this.

Nickel and Dime

The OUTPUT command can be used any time we want to change a number into a name. For example:

```
TO NICKEL
OUTPUT 5
END
```

We can combine our NICKEL with the names of other coins.

```
TO DIME
OUTPUT 10
END
```

This allows us to use Logo with word problems.

```
?PRINT NICKEL + DIME
15
```

Listful Thinking continued

It's in the Stars

You and your students may already be gazing at the nighttime sky occasionally, in anticipation of Halley's comet this spring. You may be working on a science unit about stars, in order to prepare them for the big event. Now that you've been properly introduced to the OUTPUT command, let's use it to make a Logo star map.

Some versions of Logo, such as Apple Logo and Apple Logo II, have a built-in command, DOT, which will put a dot on the screen. In other versions, such as Terrapin Logo, it's easy enough to make your own procedure that does the same thing. (If you are using Terrapin Logo, type in the DOT procedures at the end of the column before you go on.) The DOT command works like this:

```
?DOT [50 0]
```

This command should have placed a dot on the graphics screen 50 spaces from the center. Type this command in. What does it do?

```
?DOT [0 50]
```

You should see a DOT on the screen 50 spaces *above* the center. The DOT commands takes a list of two inputs. (You knew that we would get around to lists sooner or later!) The first item in the list gives how far over the dot is placed. The second item in the list tells how far up the dot is placed. What happens if you use negative numbers?

```
?DOT [0 -50]
```

Be sure that the minus sign is next to the 50, without a space in between. You can use the DOT command to make a starmap. Here is the beginning of the Big Dipper.

```
?DOT [-120 40]
?DOT [-55 70]
?DOT [-20 55]
```

Instead of typing out the star positions one at a time, we are going to make a list of star positions. First, we will put three stars in this Big Dipper procedure. We'll add the the rest later.

```
TO BIG.DIPPER
OUTPUT [ [ -120 40 ] [ -55 70 ] [ -20 55 ] ]
END
```

We'll use a modified version of a procedure called INSPECT to sprinkle the stars across the sky. INSPECT prints the first item in a list, then prints the next one, and so on until the list is empty. (If you are using Terrapin or Commodore Logo, omit the brackets around the word STOP.)

```
TO INSPECT :LIST
IF :LIST = [ ] [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

Here's how INSPECT could be used to inspect a shopping list.

```
?INSPECT [ EGGS MILK BREAD ]
EGGS
MILK
BREAD
```

Of course, we could create a procedure with a shopping list in it.

```
TO GROCERY.LIST
OUTPUT [ EGGS MILK BREAD ]
END
```

Now we don't have to type out the list each time we want to use it.

```
?INSPECT GROCERY.LIST
EGGS
MILK
BREAD
```

We can also use INSPECT to examine the list of star positions in the Big Dipper.

```
?INSPECT BIG.DIPPER
-120 40
-55 70
-20 55
```

Sprinkle a Star Map

If everything is working OK so far, you're ready to create a procedure to sprinkle the stars across the sky. The SPRINKLE procedure is exactly like INSPECT except for one change. It uses DOT instead of PRINT.

```
TO SPRINKLE :LIST
IF :LIST = [ ] [ STOP ]
DOT FIRST :LIST
SPRINKLE BUTFIRST :LIST
END
```

Listful Thinking continued

Now try the SPRINKLE procedure with the Big Dipper list.

```
?CLEARSCREEN
?SPRINKLE  BIG.DIPPER
```

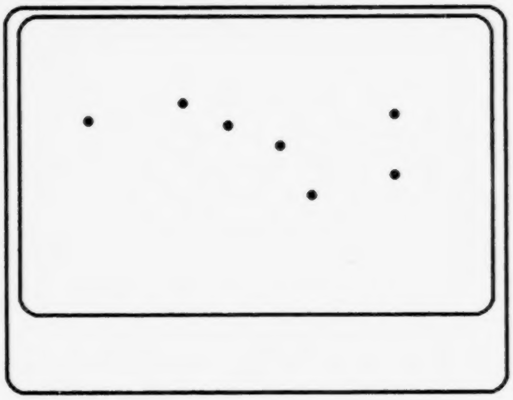
If everything worked, the SPRINKLE procedure should have sprinkled the first three stars of the Big Dipper across the screen. If that happened, we can go on to add the other four stars in the Big Dipper.

When you edit the BIG.DIPPER procedure and add the rest of the star positions, don't press the RETURN key when you reach the end of the editor line. Instead, keep on typing until you reach the end of the list in the procedure below. In many versions of Logo an exclamation point will appear as you go past the end of the editor line. This shows that Logo thinks of the two rows of numbers as part of the same line.

```
TO BIG.DIPPER
OP [ [-120 40][-55 70][-20 55][30 40][55 -10][120 5][120 60] ]

END
```

Try out the complete BIG.DIPPER.

```
?CLEARSCREEN
?SPRINKLE  BIG.DIPPER
```



Now that we have the Big Dipper up in the sky, we're ready to make the stars twinkle. Before we can make all the stars twinkle, we need to make a single star blink.

How can we make a star blink? First we turn off the star; then we turn it back on again! Here's how to turn off a star. (In Terrapin Logo and Commodore Logo substitute PC for SETPC.)

```
?SETPC 0 DOT [-120 40]
```

If that turned off the star, you can turn it back on by SETPC (or PC) once more.

```
?SETPC 1 DOT [-120 40]
```

Twinkle Twinkle

We can combine both sets of instructions (off and on) into a single BLINK command. (If you use Terrapin or Commodore Logo, substitute PC for SETPC, and omit the WAIT command.)

```
TO BLINK :STAR
SETPC 0 DOT :STAR
WAIT 10
SETPC 1 DOT :STAR
WAIT 10
END
```

Here's how to use the BLINK command.

```
?BLINK [-120 40]
```

Did the star turn off, and then back on again? Now that we have a way to make a single star blink, let's make a whole starfield twinkle. To do that, we'll randomly pick a star and make it blink. Then we'll randomly pick another and do the same thing. As we do that over and over, the starfield will twinkle.

From past columns we have a procedure to pick an item from a list. (If you are using Version 1 of Terrapin Logo, you will need additional procedures to make PICK work. These additional procedures are found at the end of the column.)

```
TO PICK :LIST
OP ITEM (1 + RANDOM COUNT :LIST) :LIST
END
```

We can use PICK to pick an item at random from a list.

```
?PRINT PICK [ JOHN SUE MARY SAM DAVE ]
 SAM
```

The same principle can be used to pick a star at random from the Big Dipper.

```
?PRINT PICK BIG.DIPPER
55 -10
```

Since PICK chooses randomly, the star coordinates it chooses for you may be different from the

Listful Thinking continued

ones it chose for us. Try this several times. If you are satisfied that PICK is randomly choosing stars from the list, you are ready to use it with BLINK.

```
?BLINK PICK BIG.DIPPER
```

Now we can make a TWINKLE command.

```
TO TWINKLE :STARS
REPEAT 100 [ BLINK PICK :STARS ]
END
```

To see the stars twinkle, just type:

```
?TWINKLE  BIG.DIPPER
```

You should see stars twinkling! Make a Logo wish! We'll see you next month.

Terrapin Logo Procedures

Dot Tools

To compensate for the aspect ratio of the video monitor, the turtle skips every fifth step when it is traveling vertically (2, 7, 12, 17 22, etc.) and does not make a dot. The DOT command in Apple Logo compensates by moving up one turtle step at those positions. In other words, if you specify DOT [10 12] in Apple Logo, the dot actually is placed at [10 13] instead. The following set of Dot Tools provide a dot function in Terrapin Logo. They compensate for the problem of dots falling "in the cracks" every fifth step by using an approach equivalent to the one employed in Apple Logo. That is, if DOT [10 12] is specified, the DOT actually appears at DOT [10 13].

```
TO DOT :POSITION
PU HIDETURTLE
SETXY FIRST :POSITION LAST :POSITION
IF IN.CRACK? FD 1
PD POINT
END

TO IN.CRACK?
OUTPUT LOCATION = INTEGER LOCATION
END

TO LOCATION
OUTPUT ((ABS YCOR) - 2) / 5
END

TO ABS :NUMBER
IF :NUMBER < 0 OP -1 * :NUMBER
OP :NUMBER
END

TO POINT
FD 1 BK 1
END
```

Procedures required for PICK with Version 1 of Terrapin Logo.

```
TO PICK :LIST
OP ITEM ( 1 + RANDOM COUNT :LIST ) :LIST
END

TO ITEM  :NUMBER  :LIST
OUTPUT ITEM.LOOP  :NUMBER  :LIST 1
END

TO ITEM.LOOP :NUMBER :LIST :COUNT
IF :LIST = [] OP []
IF :NUMBER = :COUNT OP FIRST :LIST
OP ITEM.LOOP :NUMBER BF :LIST :COUNT + 1
END

TO COUNT :LIST
OP COUNT.LOOP :LIST 0
END

TO COUNT.LOOP  :LIST  :NUMBER
IF :LIST = [] OP :NUMBER
OP COUNT.LOOP BF :LIST :NUMBER + 1
END
```

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a doctoral student at the University of Virginia. She is interested in Logo applications in language arts and speech-language pathology.

BACK 10

OOPS!! In the NLXtra last month, we announced a number of Logo related publications now available from the London University Logo Group, but neglected to include their address! Here is the entire notice reprinted, with the necessary address.

Celia Hoyles, of the London University Logo Group, announces the availabilty of the following publications on Logo. (All prices are listed in English pounds due to the uncertainty of the monetary exchange rates.) *The Interim Report on the Logo Mathematics Project 1983-4*, £8.50 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Proceedings of the Logo and Mathematics Education Conference, March 1985*, £5.00 plus postage and packing: £7.06 air mail or £2.10 surface mail. *Creating a Mathematical Environment through Programming: A Study of Young Children Learning Logo*, by Richard Noss, £10.00 plus postage and packing: £10.00 air mail or £5.00 surface mail. Celia Hoyles, London University Logo Group, University of London Institute of Education, Department of Mathematics, Statistics and Computing, 20 Bedford Way, London, England WC1HOAL.

NLXionary
A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"Exciting Effects of Logo in an Urban Public School System," by Peter Fire Dog, *Educational Leadership*, September 1985.

>The author reports on the research conducted during the three years of one of the largest Logo efforts in the ocuntry. (A total of 250 K-12 teachers are currently involved.) The project attempts to have teachers use Logo in their classrooms to engage all types of students in learning. Typically, this means one or two computers in a corner of the classroom occupied all day long by pairs of students while regular instruction continues. This averages out to be approximately 40 hours of Logo per student per year.

>The research suggests that (1) different kinds of social, emotional, and congitive effects of Logo develop over time, and (2) classroom environmental factors related to student learning are many and complex. Not exactly eye-popping revelations, but a closer look at the data is a bit more interesting.

>The top effects when measured in January of 1984 were:

1. enthusiasm about learning
2. sharing, teaching, consulting
3. frequency of interaction with peers
4. logic, structured thinking, problem solving
5. likableness, sense of acceptance, belonging

>In May of the same year, the top effects were:

1. likableness, sense of acceptance, belonging
2. sharing, teaching, consulting
3. logic, structured thinking, planning
4. pleasure in work
5. frequency of interaction with peers

>But those effects that showed the greatest *improvement* from January to May were:

1. spelling and writing
2. rate of learning
3. homework, class assignments
4. level of achievement
5. reading, reading comprehension

>I would have liked the author to comment more on this latter phenomenon of Logo impacting spelling, writing, and reading. It may have some parallels to Stanley Pogrow's *Higher Order Thinking Skills (HOTS)* project in Arizona (reported in the same issue of *Educational Leadership*) in which Chapter I students make significant gains in basic skills through the use of problem solving / thinking skill software ... especially since Pogrow has frequently made critical comments about Logo.

>A more important finding, I believe, is that an analysis of student subgroups revealed no patterns for significant improvement. In other words, in comparing the lowest and highest achievers, males and females, working vs. middle class, socially accepted vs. isolates, etc., the rates of improvement were about the same.

>Likewise, none of the usual student characteristics (disruptiveness, initiative, independence, cognitive style, emotional development) typically associated with student academic success or failure were found to be predictive of who will show major improvement with Logo. Fire Dog sums it up best:

"From a research perspective, the tentative conclusion is fairly straightforward: Logo effects seem to be both wide ranging and substantial, and appear to be available to students from almost any type of learning, social, or motivational background. Logo clearly represents an innovative way to create opportunities for academic success and enrichment for most students."

(Ed. note: Peter Fire Dog also presented a talk entitled "Logo Effects in Public School Classrooms," at the Logo 84 conference at MIT. For more information, write to Peter Fire Dog, University of Minnesota, Department of Sociology, 267 19th Avenue South, Minneapolis, MN 55455.)

"Pulling in the Reins on Freewheeling Logo," by Alan Alterman, *Classroom Computer Learning*, September 1985.

>This is a practical follow-up to the article I highlighted last month, "Logo Today: Vision and Reality," by Uri Leron, in *The Computing Teacher*, February 1985. Alterman explains five of his strategies for structuring the teaching of Logo in a one computer classroom while simultaneously fostering natural exploration and discovery. While most of his approaches will not be new to long-time Logo teachers, they do provide a tidy summary for the teacher newer to Logo. His classroom management strategy is the best I've come across.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

TurtleTips

by
Jane Toth and
Donna Lanyi

Turn your classroom into an instant mass production facility! Although primary students can master Logo primitive commands, we've found that they aren't capable of the time commitment necessary to produce fully developed pictures. Primary students need immediate success and gratification. Susan Wood from St. Francis School in Goshen, Kentucky, sent us some excellent ideas to speed up picture completion for younger students, K - 2.

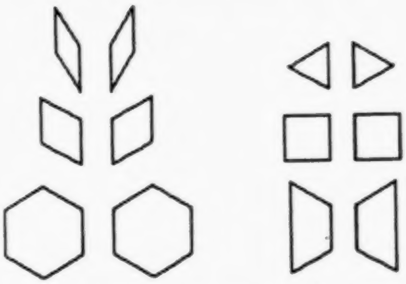
The Turtle Teacher

Susan has designed her own version of an "E-Z Logo." She felt that her children needed to relate programming to the physical world around them. Using pattern blocks seemed natural because of their popularity, invitation to creativity, and adaptability to the computer. The children have long been familiar with pattern blocks, and how they fit together to make beautiful designs.

Susan wrote procedures for each pattern block piece. She included a left and right drawing procedure for each, to make moving the pieces on the computer easier.

Next, Susan drew each pattern block on paper, showing the turtle's heading and the procedure name for each. She taped these drawings on the table beside the computer for easy reference.

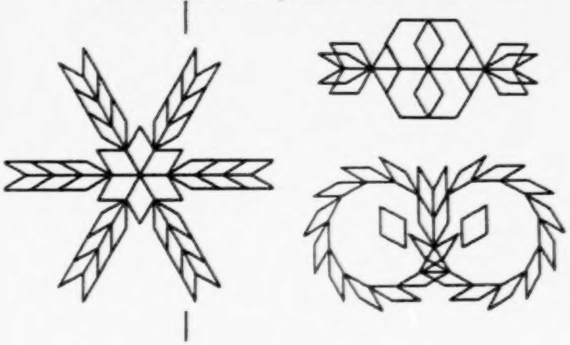
To help the children fit the pieces together easily, she decided to use the same size for each side of the blocks (FD 25 or BK 25) and to use multiples of 30 for turns. The long side of the trapezoid was 50, the only "non-25" side of any figure.



Only four turns were needed for fitting the pieces together: LT or RT 30, LT or RT 60, LT or RT 90, and LT or RT 120. The children could choose one of the turns from this list. If their choice was not the "right" one, they could undo it by typing its opposite turn, and then try one of the remaining angles. They not only learned to undo, but also to make excellent choices.

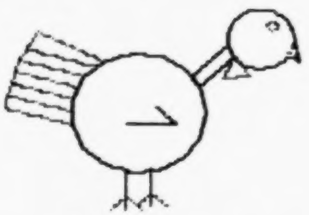
Susan had two children work together, so that, while one was making the design, the other was advising and recording in their journal. They used the journal as an aid in making the design into a procedure to be saved. From the journal, she could also help students discover how they could use REPEAT to shorten the procedure. Saving the procedures provided the basis for an on-going project in which they kept adding on to their designs.

Here are some samples of her students' work.



Procedure of the Month

Start your class early in November creating the first Thanksgiving. Have students design and draw figures to post on a Thanksgiving Dinner scene for a bulletin board. Use our turkey procedure to create Thanksgiving cards to be taken home, or have the students create their own "Turkey Cards," perhaps using the pattern block procedures.



If you are interested in a listing of the turkey procedure and / or Susan's pattern block procedures, please send us a self-addressed stamped envelope.

We invite any teachers or elementary students to send Logo procedures or ideas to be shared in this NLX column. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher to Teacher

A Review of New Logo Materials

By James Fry

This month's reviews are on some products that address an area of long time interest to Logo. People have been asking about "microworlds" and what they are, but publications have not addressed this issue until recently. Here are comments about two recent products which provide teachers with microworld-like activities for their students.

Logo Microworld: Picture Puzzles, by Ihor Charischak, 1985, published by K - 12 MicroMedia Publishing, Inc., 6 Arrow Road, Ramsey, NJ 07446, (201) 825-8888, $19.95.

Target Audience: Upper elementary children with Logo experience

Content: *Logo Microworld: Picture Puzzles* comes with a booklet and a disk. The booklet is divided into an introduction, a section on use in an instructional setting with directions for the accompanying disk with your Logo version, and five puzzle sheets to go along with the five lessons or microworlds.

The puzzle sheets contain goal statements, drawings, hints, and suggestions for further exploration. The materials are intended for students who are familiar with Logo procedure writing and editing. The puzzle sheets can be reproduced to be given to the students.

Strengths: *Logo Microworld: Picture Puzzles* is described as " a package of five programs designed as a learning environment using Apple Logo or MIT Logo". If you are looking for microworlds then this package is something you would like.

The strengths of *Logo Microworld* are many. Most importantly, it encourages students to write well organized and structured programs. The activities are set up to make the students think about and make use of a modular approach. This should also help teachers with little or no programming background to learn more about structured programming.

As an example of how the lessons are organized, consider Lesson #1. The goal is to write a procedure to draw a "happy face." The students are given 3 Picture Piece procedures and 3 Move procedures to position and move the turtle. The challenge is to figure out the proper order of the 6 procedures to make the face.

As the students move through the lessons, the tasks become more complex. In Lesson # 5, they are given 4 Picture Pieces and must write their own MOVE procedures to produce a picture of a neighborhood.

Suggestions: I encourage the author to develop more microworlds of this nature.

KNOW LOGO, A Logo Exercise Program, by George Bredehorn and Learning Well, 1985, published by Learning Well, 200 South Service Road, Roslyn Heights, NY 11577, (800) 645-6564, $49.95.

Target Audience: Students in grade 1 and up

Content: *KNOW LOGO* comes with a guide book and disk. There are two separate versions, one for Terrapin and one for Apple Logo. *KNOW LOGO* consists of over 50 learning activities for exploring areas of Logo. The activities are organized by the types of skills that they reinforce, such as concepts of direction, distance, angles, and circles and arcs. The guide book consists of an introduction, how to work with the disk, and a brief description of the programs on the disk.

Strengths: For a teacher just beginning with Logo or a teacher with a group of inexperienced Logo students, *KNOW LOGO* offers a variety of activities to explore and expand upon. For example, the program called Artist1 consists of 8 different objects from space, including a rocketship, a star, and an astronaut. All the objects can be drawn in many different sizes by using an input number with each name. Students can then develop their own sets of procedures or pictures. This idea could be used in a classroom where students could put together a collection of all types of shapes to be shared by each other.

Suggestions: I would like to see the guide book expanded to give more ideas for extended activities and suggestions for teacher use in the classroom. I also suggest hat a price of $49.95 seems rather high for Logo activites with such limited documentation.

The Logo Information for Teachers (LIFT) group is interested in hearing from teachers who can identify new materials and books they have found useful in teaching Logo. Please send your suggestions and comments to: LIFT, PO Box 5396, Plymouth, MI 48170.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI, and is a co-founder of the Logo Information for Teachers (LIFT) group.

Q and A

by
Jim
McCauley

Q. Logo is supposed to be easy to learn. My first lessons were easy enough, but now I'm finding that Logo is harder to learn than I imagined. Is this the usual case?

A. It certainly was for me! I began learning Logo when it first came out on the TI 99/4 computer, and I spent most of the first few months learning what my friend James Milojkovic calls "spinal cord Logo." That is, I mucked about doing clever things with the turtle and the TI's wonderful sprites. I did nothing with list processing until I picked up Apple Logo. Then I hit a brick wall.

I had done some fairly complicated things in other computer languages (BASIC, Pascal, PL/1, and Forth), but nearly all these projects had been based on the processing of indexed arrays (See last month's Q & A). I didn't know a linked list from a hole in the wall, and had never formally studied data structures.

The Simple Truth

Frankly, I felt intimidated by the structured data types of languages more complicated than BASIC, and that was one of the things that led me to Logo. In Logo, after all, the only objects to deal with are words and lists. That kind of apparent simplicity was right up my street.

I had, of course, completely misconceived the "simplicity" of Logo. The absence of indexed arrays in nearly all implementations of Logo essentially forces the use of recursion to decompose, traverse, and recompose data structures built from lists. I had done almost nothing with recursion in other languages, and there were no Logo books to explain it in the spring of 1982.

That "Help, Lisp!" Feeling

So, I turned with fear and trepidation to Lisp, Logo's predecessor. I had heard that Lisp was arcane, complex, and subtle, and was understandable only by freaky hackers at MIT. In the hope that these rumors were not entirely correct, I bought a Lisp interpreter and several Lisp books.

Most of the books were just awful (I refuse to read them even now, unless absolutely forced by circumstance), but I found two gems among the dross: *Artificial Intelligence*, by Patrick H. Winston and Berthold K. P. Horn (Addison-Wesley), and *The Little LISPer*, by D. P. Friedman (SRA Publications). Careful study of these books (and much hacking into the wee hours) gave me a Lisp perspective I could carry back to Logo.

Logo List Ladders

Today, though, learners of Logo who hit "the wall" are more fortunate. There are several Logo books that can help you get over, under, or around the conceptual blocks you face in understanding the list processing understructure of Logo. Here are two that I believe are especially useful.

- *The Second Logo Book: Advanced Techniques in Logo*, by Dan Weston (Scott, Foresman and Co.). This book contains an especially clear introduction to list processing concepts and a large part of what I brought back to Logo from my study of Lisp.
- *Computer Science Logo Style: Intermediate Programming*, by Brian Harvey (MIT Press). This is an excellent source for a solid foundation in list processing and advanced ideas in program design.

Finally, I would encourage serious Logo students to get acquainted with Lisp. A very good introductory text is *LISP: A Gentle Introduction to the Art of Symbolic Computation*, by David S. Touretzky (Harper and Row).

This monthly column features questions about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to me at 876 East 12th Street, #4, Eugene, OR 97401. Please enclose a self addressed stamped envelope.

Jim McCauley is a graduate student at the University of Oregon, studying wtih David Moursund, and has written Logo articles for many national publications.

NLXual Challenges

by Robs Muir

Ella and the n-gon

If the Lord Almighty had consulted me before embarking on the Creation, I should have recommended something simpler.
-Alfonso X, King of Castile (1226 - 1284)

There is a rumbling in the school hallways that is growing to a crescendo. Depending on the region, this song is called, alternately, problem-solving, critical thinking, or critical reasoning. Now, every "Educational Consultant" smart enough to hang a shingle is jumping on the bandwagon to sell the best package to improve your students' thinking skills; reproducible worksheets and diskettes optional. However, like all good music, this tune can be played on the commonest of instruments.

Sorry to harp on about all this, but this whole fascination with problem-solving sounds like a round which repeats itself in Education every 20 years or so.

NLXual Challenges continued

You don't need any special software or hardware to create a healthy problem-solving environment in your school or classroom. What you really need is a terminal case of over-enthusiastic curiosity about the world and a desire to "find out." This is bound to be infectious.

Circle and the Square

You may remember an interesting problem in this column last year about circles and boxes (See NLXual Challenges, January 1984 NLX). The challenge was to write a Logo program that would draw a circle within a square of any given size such that the circle is tangent to the sides of the square. Stop here a moment! Can you do this?

We have gotten a lot of life out of this challenge, spurred, by the fact that several of my colleagues unwittingly derived the value for pi in the course of their explorations. You learn something old every day!

I happened to give this challenge to some of my graduate students two summers ago and one student came back with a challenge of her own, a challenge that still baffles me. Ella Pfeiffer gave this CIRCLE.BOX challenge a new twist that may give some problem-solver devotees food for thought. Can you solve this one?

Circle and the n-gon

Try this: Get the turtle to draw any n-sided regular polygon. Next, get the turtle to draw a circle such that it lies inside the polygon. Yes, of course, it must be tangent to the polygon! Next, generalize this algorithm to work with *any* regular polygon. If you have some difficulty with the math as I did, dig out those old 8th grade geometry texts.

After several weeks, Ella solved this problem that she had imposed on herself and she proudly demonstrated her handiwork to the class. We watched in silent awe as the turtle appeared to do her bidding. Ella also proceeded to explain, in great analytical specificity, her arrival at the "truth."

Since you've solved this problem (right?), here is Ella's solution.

```
TO POLYCIRC :S :L
;[ INSCRIBES A CIRCLE WITHIN A POLYGON ]
;[ WRITTEN BY ELLA PFEIFFER ]
;[ FIRST DRAW THE POLYGON, THEN THE CIRCLE ]
REPEAT :S [ FD :L RT 360 / :S ]
FD 0.5 * :L
REPEAT 360 [ FD ( :L * 11 * COS (180 / :S ) ) / ( 1260 *
      SIN (180 / :S ) ) RT 1 ]
END
```

Now, if you appreciate Nature's simplicity, you might have reason to question this solution. Where did the 11 come from? Why the number 1260? This is a far cry from the simple elegance of the Total Turtle Trip Theorem! Yet, Ella Pfeiffer claims that this is mathematically correct and can be analytically derived.

This month's challenge then: Can you prove or disprove Ella's claims?

I'm afraid that, like Ptolemy before me, I'll spend years vainly seeking to prove that the Universe is simpler than evidence might suggest. And that 11's and 1260's aren't a part of the Turtle's Trip.

Send your analysis and conclusions, along with a self-addressed stamped envelope to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, CA 91711

We will redistribute to all contributors a representative cross-section of responses.

Do you have a favorite Logo challenge to share with other NLX readers? If so, then send it to the above address.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.

The National Logo Exchange, copyright © 1985 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717.

Editor..........Tom Lough

THE NATIONAL
LOGO
EXCHANGE

The Logo Newsletter for Teachers

Volume 4 Number 4 FORWARD 100! December 1985 Page 1

Come to the LogoShow!

by Doris Schroeder

"Hello! Welcome to the LogoShow!"

"My name is Julie, and this is Michael. We will tell you the story of *The Gingerbread Man.*"

A title page flashes off the computer screen and you are treated to a Logo drawing showing the little old woman's view of the gingerbread man in the center of a pan. As the story unfolds, two fourth grade students retell the familiar tale to a rapt audience.

At the appropriate moments, Julie presses the <RETURN> key, and new Logo illustrations appear: the gingerbread man running further and further down a road, the gingerbread man riding the crest of the waves on the fox's back, the gingerbread man's head just above the water, and in the final scene, only the blue waves.

Michael concludes, " ... and before they got to the end of the river, the fox ate up the gingerbread man. And that is what should happen to all gingerbread men."

A look around the gym shows equally inventive Logo storytelling.

How It All Began

In previous years, the fourth grades at Montclair-Kimberley Academy, a private K-4 elementary school in Montclair, NJ, presented traditional-style class plays. This year, however, Kristina Urband and I got an inspiration that our fourth graders might be interested in something different ... The LogoShow! Although our students had already worked with Logo a number of weeks, the results of these few moments of inspiration were more Logo lessons, and many, many hours of hard work.

Each pair of students selected a fairy tale to retell and illustrate with Logo. They designed their pictures on paper, then created them with Logo on the screen. They wrote the stories and sketched storyboards in "dummy" books, sharing the narration, revising, and editing as needed. They rehearsed at the computers, receiving assistance from their teachers and peers. And, for the finishing touches, they dressed up as characters from their tales and presented their stories to the rest of the school at the LogoShow!

Sixteen Apple computers with color monitors were arranged, two to a table, around the gym. The school's two hundred students and many parent-visitors were divided into eight groups and assigned a starting position. After both stories at a table were completed, a student raised a turtle sign. When eight turtle signs were held high, a few notes from the xylophone signaled everyone to rotate one position clockwise to hear the next two stories. The assembly proceeded in this fashion until all sixteen stories had been presented to each group. It was a phenomenal exercise in coordination and planning, and quite a success! And our students were the most excited of all!

It was also a phenomenal exercise in coordinated learning. In addition to all the Logo and language arts activities that were highlighted, one girl adapted the Hall and Oates song, "Out of Touch," to go with the show.

In nineteen years of primary school teaching, I thought the LogoShow was the most spectacular and exciting learning project I have ever carried out. It gave everyone a chance to get involved. I heartily recommend it for others!

(Ed. Note: This would be an outstanding Logo class project for National Library Week in April. Doris is interested in corresponding with any NLX readers who want more details about the LogoShow. Write to Doris

LogoShow continued

Schroeder, Montclair-Kimberley Academy, 201 Valley Road, Montclair, NJ 07042. Please enclose a stamped self-addressed envelope.)

Doris Schroeder taught fourth grade at Montclair-Kimberley Academy in Montclair, NJ, at the time of the LogoShow. She is presently teaching seventh grade reading and writing, and chairing the Academy's middle school computer department.

Out of Space

A Logo adaptation of
Hall and Oates' "Out of Touch"

by Margo Hirsch, fourth grade
Reproduced with permission

I look down at the keys
And then I press Control C.
Then it said, "OUT OF SPACE."
I wanted to play that turtle game RACE.
I looked up at the TV screen,
Took out the disk, shut off the machine.
It's no use. Why does it happen to me?

Chorus

You're OUT OF SPACE.
RECYCLE NODES.
But I still love to use Apple Logo.
RECYCLE NODES.
But I still love to use Apple Logo.
Oh-oh-oh, Lo-oh-go, Lo-oh-go.

Put Apple Logo in the disk drive
To get that computer to come alive.
It's booting up and it's ready to go.
Type S and T so the turtle will show.
It shows the turtle standing in place.
I'll type a procedure to make a face.
FORWARD 50 RIGHT 90 PENDOWN

Chorus

Chorus (at the end of the LogoShow)

We're OUT OF TIME.
It's time to go.
But we hope you've enjoyed our LogoShow!
We're OUT OF TIME.
It's time to go.
But we hope you've enjoyed our LogoShow.
Oh-oh-oh, Lo-oh-go, Lo-oh-go.

From The Editor

by Tom Lough

"Sorry. We cannot approve this course for recertification. Logo isn't appropriate for use in the high school."

Because Logo is powerful enough that it can be configured for use with young children, it is often misconstrued as a computer language only for lower grades. After all, most of the Logo introduction in Papert's *Mindstorms* was directed toward use with primary and elementary children. The fact that an entire chapter is devoted to Logo and the teaching of physics is often overlooked.

As a result, a number of administrators (and experienced Logo teachers) can lose sight of the fact that this artificial intelligence language is powerful and sophisticated enough to be used at all levels of the educational spectrum. This mindset is not to the advantage of our students.

Statements such as the hypothetical quote above do little to encourage the development of Logo applications in the upper grades. And the problem may be getting worse.

Elementary teachers might be tempted to consider the "no Logo in the high schools" situation as something which belongs to their secondary counterparts, and choose not to become involved. If this happens, then the value of the Logo learning in the elementary grades can legitimately be called into question. What is it worth if no one from the lower grades is interested in the follow through at the higher levels?

It would be as bad, however, if elementary teachers give the impression that they feel Logo "belongs" only in the lower grades. In many cases, where administrators might be considering supporting experimental Logo projects in the high school curriculum, the gears of their minds could become locked up by thoughtless remarks made by well-meaning elementary teachers extolling the virtues of Logo for their grade levels with no regard for Logo applications at higher levels.

Perhaps if we all become more sensitive to the full potential of Logo and more interested in the big picture of Logo applications throughout the K-12 (and beyond) curriculum, our students would be in a position to take advantage of the power Logo offers throughout the entire grade sequence.

If we do not, high school students who learned what Logo can do while in elementary school might begin asking some rather uncomfortable questions ... While that may not be bad in itself (... and a child shall lead them ...), wouldn't it be better if Logo were already understood, respected, and "approved" by the administrators involved?

What do you think about it?
What can we do about it?
FD 100!

Tom

NLXtra

A Supplement to the National Logo Exchange Newsletter

Volume 4 Number 4 *FORWARD 100!* December 1985 Page 1

Online from Paradise: World Logo Conference Report

by Sandy Dawson

What do you get if you invite 90 Logo people to a retreat centre in the Canadian mountain wilderness 85 kilometres north of Vancouver, British Columbia? What if you then give them two computer terminals connected via phone lines to the CompuServe computer in Columbus, Ohio, and arrange for their calls to be managed by a man in a garage in Richardson, Texas? What if Logo people from all over the world start calling in? People from England, Spain, and Israel. People from Iceland, Denmark, and Australia. People from Boston, New York, Chicago, St. Louis. People from Keene NH, Carmel CA, Eugene OR, Newton MA, Ellison WI, and Charlottesville VA, to say nothing of Canadian cities like Kingston, Bathurst, and Vancouver itself.

What you get, of course, is the **World Logo Conference!**

In addition, if you arrange formal working sessions onsite, plus the time for impromptu gatherings and discussions, mix that with candlelight dinners, instruction in and debugging of "contra" dancing (a form of New England line dancing imported via Dan Watt to Vancouver), astound folks with the swirl of the Scottish "pipes" accompanying a highland fling, and top it off with a cake commemorating the first birthday of the British Columbia Logo Users Group, then you have the makings of one heckuva celebration!

The World Logo Conference took place October 25-27, 1985. Gerri Sinclair and Sandy Dawson, conference co-chairs, initiated the online conference with an electronic wine and cheese, chatting with Jim Muller, systems operator of CompuServe's Logo Forum, and the man in the Texas garage. The conference was launched, and to the amazement of many, everything was working. For, you see, a conference such as this had never been done before. No one, to our knowledge, had tried to run simultaneous onsite and online conferences; nor had anyone tried to run such a geographically diverse conference as was being attempted here. And it all worked! The pace of the conference accelerated quickly.

As onsite participants watched the premiere of a new Logo film called, "Beyond Turtle Graphics: Growing Up with Logo," online participants partook of discussions with Tony Stavely from New Hampshire, Jim Muller from Texas, and Brian Silverman, Harold Brochmann, and Molly Watt from onsite, closing off the evening of the 25th with the Logo nightowls.

Seymour Papert, scheduled to be the featured onsite speaker the next day, telephoned in ill, but arrangements were quickly made for him to host a session online from Boston. With Seymour's presence being monitored around the world and throughout the Paradise site, the conference shifted into overdrive. By Saturday night's candlelight dinner, the participants had been stimulated to explore "technocentricity," Logo for the able and not so able, Logo dilemmas, ceilings and thresholds, and had been challenged by researchers and workers such as David Thornburg, Tom Lough, Dan Watt, Dale Burnett, Joel Hillel, Rina Cohen, Bill Higginson, and Tom Kieren.

Onsite, local users and learners of Logo shared their experiences and expertise. Members of Logo user groups from St. Louis, Chicago, London, Australia, and New Hampshire contributed their enthusiasm and critical concern for the extension of Logo into new areas.

Molly Watt (an onsite presenter and godmother to the BC Logo Users Group) commented about the conference: "When I was online talking to people around the world, it was so exciting that I didn't want to be bothered with the onsite program. But, when I got involved in the onsite activities, I couldn't have cared less about the online stuff! It was so *hard* to decide which part to choose and which part to leave out."

On Sunday morning, Uri Leron appeared on the network from Israel right on schedule. The following five hours flowed by like the rushing waters of the Squamish River nearby, with Uri, Dr. Logo (Joe Power), Jim McCauley, and Paul Goldenberg blending their discussions together with online participants to concoct a rich, satisfying drink of Logo extensions and criticisms. Meanwhile onsite, Brian Silverman was talking about

World Logo Conference continued

"nothings" as he made his Life Game dance across Logo monitors.

But what about outcomes? Well, we know the onsite people experienced a tremendous "high," and that an environment which was warm, supportive, self-correcting, and non-threatening was created at Paradise. We know that, although some folks had difficulty getting online, they really enjoyed the conference, as evidenced by their own expressions in the online transcripts. We also noted that such a conference can be run literally from anywhere in the world, because Paradise Valley is *remote!*

But I've said nothing about Logo! All of the above is about a Logo environment. Paradise Valley was a model for learning Logo. Did we learn more Logo? Probably not; but what we did learn was how to help others learn and teach with Logo. We explored some extensions of Logo, as well as some limits.

We met one another, and perhaps that is the most important thing of all. Though this conference was very "high tech" in nature, it also demonstrated the power of the teleconferencing medium for fostering "high touch" interactions. We saw Gerri interrupt the British Logo Users Group online from England to ask them to run downstairs to answer the phone so she could give them instructions on how to help a group from Iceland make their online connections. Molly Watt chatted online with her daughter in an apartment in New York's lower east side. Joel Hillel, Rina Cohen,and Ofer Arad (all onsite) and Paul Goldenberg (online) created an emotional Israeli connection with Uri Leron.

These "high touch" memories linger long after the conference is over. Ultimately, that is what Logo is all about, isn't it?

Sandy Dawson is a member of the faculty of Simon Fraser University in North Vancouver, British Columbia.

Logo Notices

How about a turtle you can ride? Innovations offers such a toy on page 26 of their current catalog! And it is green and triangular shaped, too! To make it go, you just sit down on it and move the handles. For a copy of the catalog, write to Innovations, 110 Painters Mill Road, Owings Mill, MD 21117, or call (800) 638-6170 or (800) 228-5000. To order the Flying Turtle, specify item number 15537, and send $39.95 plus $3.95 for shipping and handling.

The Virginia Department of Education has produced a 30-minute videocassette on programming and problem solving with Logo as part of a 16 part series of instruction with microcomputers. Based on a script written by Steve Tipps, the Logo video features a number of Virginia Logo users and examples of their classroom work. A 90-page activity guide for the entire series is available with the video. For more information, write to Pete Pantsari, Southern Educational Communications Association, PO Box 5966, Columbia, SC 29250, or call (803) 799-5517.

Microquests is a quarterly Logo publication dedicated to mathematics explorations. Authors Kathleen Martin and Donna Bearden are experienced Logo educations, and do a masterful job with each contant-packed issue. For more information, write to Microquests, Martin-Bearden, Inc., 2502 Peach Tree, McKinney, TX 75069.

Brian Schott of the Decision Sciences Department of Georgia State University, has sent listings for Commodore Logo procedures which allow a rather fast graphics dump to selected printers within Logo. Brian has tested the procedures with a C.Itoh 8510 printer, but feels that they should work on most Epson and Gemini printers as well. He is willing to furnish copies of these listings at no cost to Commodore Logo users. For a free copy of these listings, send a self addressed *stamped* envelope to: NLX Commodore Logo Listing, PO Box 5341, Charlottesville, VA 22905.

ExperTelligence announces *French ExperLogo* and *ExperLogo-Talk* for the Macintosh. For more information on these products, write to ExperTelligence, 559 San Ysidro Road, Santa Barbara, CA 93108, or call (805) 969-7874.

The latest Unisource catalog lists TI Logo II for $69.95. For more information, write to Unisource, PO Box 64240, Lubbock, TX 79464.

Deals on Disks

The following business firms offer economical prices on disks. Write or call for their most recent prices.

- Blackship, PO Box 883362, San Francisco, CA 94188. (800) 235-6646 ext. 31. From CA, call (800) 235-6647 ext. 31.
- Disk World!, Inc., 30 East Huron Street, Suite 4806, Chicago, IL 60611. (312) 944-2788.
- Diskette Co-op, PO Box 65816, West Des Moines, IA 50265. (515) 225-6629.
- Magnetic Media of New England, PO Box 780, Beverly, MA 01915. (800) 992-0123.
- Megasoft, PO Box 1143, Freehold, NJ 07728. (800) 222-0490.
- Unitech, 20 Hurley Street, Cambridge, MA 02141. (800) 343-0472.
- User Friendly East, PO Box 830, Point Pleasant, NJ 08742. To order, call collect (201) 892-9899.

Tipps for Teachers

by Steve Tipps

Geologoboards

Working with geoboards is a favorite student activity. It is easy and fun to make shapes, designs, and symbols using varicolored rubber bands stretched between the nails or pegs. In the process, students learn about angles, lengths, and patterns with a dash of creativity. Geoboards can also be drawn with Logo. The learning with Logo Geoboards extends understanding co-ordinate names for placement of lines and objects on the screen.

Turtle Zip

As a beginning activity, children zip the turtle about the screen using the SETX, SETY, and SETXY or SETPOS commands. Interesting or chaotic patterns result from a series of commands.

```
SETX 50 SETY -50      or     SETX 50 SETY -50
SETXY 100 100                SETPOS [100 100]
SETXY 40 (-40)               SETPOS LIST 40 -40
SETXY 50 50                  SETPOS LIST 50 50
```

A sequence of commands using the coordinate commands might be used to draw a ship, a house, or a bird. The pattern might be a geometric design. In the process of turtle zipping, students learn that the turtle can be moved directly to any spot on the screen. Students also become familiar with the range of numbers used for movement, and the syntax of the commands - especially the places which require parentheses when the second input is negative. The LCSI versions of Logo demand that the co-ordinates be in the form of a list. This can be done in two ways. The coordinates must be placed in list brackets, or the command LIST can be used for the same effect.

However, typing in the numbers each time is rather laborious. A better method of telling the turtle exactly where to go is needed. With enough experience on procedures, students should have an intuition that the same task could be done more economically with procedures. Making tools is an important process to be learned in computing. Procedure making is an experience in making tools.

A Line, B Line

The first tool creates the grid of points on the Logo Geoboard. The physical geoboard may have been labeled with letters or numbers to locate points on the grid. On a 5 X 5 grid, the points might be labeled A1, A2, A3, A4, A5, B1, B2, etc., or by AA, AB, AC, AD, AE, BA, BB, etc. If labels are not already on the geoboard, you can use tape and markers to make them.

Placing points on the screen may be done in several ways. If your version of Logo has a DOT command, it works in much the same way as SETXY or SETPOS. Simply call the DOT with two coordinates as input.

```
DOT [ 50 50 ]
```

or

```
DOT LIST 50 50
```

(Ed. note: In some Logo versions, a dot cannot be made until the pen is in a particular state, such as PD or PU.)

If you do not have the DOT command built into your Logo, you can write a DOT procedure.

```
TO DOT :X :Y
PU
SETXY :X :Y
PD FD 1 BK1 PU
END
```

or

```
TO DOT :X :Y
PU SETPOS LIST :X :Y
PD FD 1 BK 1 PU
END
```

Glen and Paula's October 1985 "Listful Thinking" column contains additional DOT procedures worth studying.

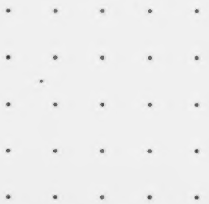
DOT can be used in the procedure for making the geoboard grid. A modular way to do this would be to setup each line of DOTs in a separate procedure.

```
TO A.LINE        TO B.LINE        TO C.LINE
DOT 0 0          DOT 20 0         DOT 40 0
DOT 0 20         DOT 20 20        DOT 40 20
DOT 0 40         DOT 20 40        DOT 40 40
DOT 0 60         DOT 20 60        DOT 40 60
DOT 0 80         DOT 20 80        DOT 40 80
END              END              END
```

Tipps continued

Finally a grid is made with the lines of DOTs.

```
TO GRID5X5
A.LINE
B.LINE
C.LINE
D.LINE
E.LINE
END
```



In the primary grades, you may choose to write the GRID5X5 procedure for the children to use as a tool. In the intermediate grades, you can show the students how to make one line and they will finish the grid. The modular nature of the grid makes it easy to enlarge. In junior or senior high, you might use this set of procedures as the crude solution and challenge students to use a list of points to improve it. See "Listful Thinking" in the November 1985 NLX for a list processing method used to make star maps.

The basic strategy is adaptable to grids of all sizes and shapes. By changing the coordinates used and the number and distance between them, the grid can be as detailed or as simple as desired. The 5X5 grid uses only positive numbers, but larger grids may force students into the other quadrants of the screen.

Beyond a 5X5 grid, keeping up with the point names is difficult. You may find that a labeling scheme is needed on the screen. The top text line of the SPLITSCREEN is handy for the horizontal labels, but you need graphics procedures or the NLX alphabet for vertical labels. Commodore and IBM Logo allow you to stamp letters on the turtle screen, which is very helpful.

Dot to Dot

With the grid in place, the turtle is ready to zip from point to point by name. Each point on the 25 point grid can be written as a positional procedure. Here are two points for GRID5X5.

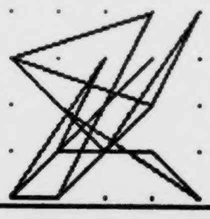
```
TO A1
SETXY 0 0
END
```

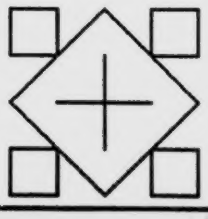
or

```
TO A1
SETPOS LIST 0 0
END
```

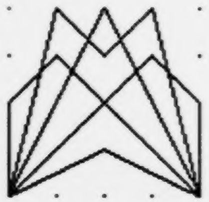
```
TO B1
SETXY 20 0
END
```

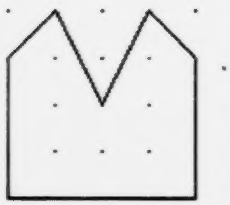
```
TO B1
SETPOS LIST 20 0
END
```





With all the zip procedures written, students are ready for creative designs. Experimentation with the tools yields different geoboard designs.

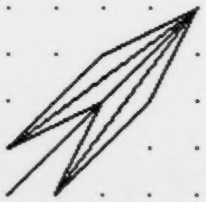




At first, the students just type in the zip procedures. They will begin to plan their designs. Working out the sequences of positions is a good off computer activity. Next come pen colors with PENUP and PENDOWN to make more complex designs. Some designs need more space and points, demanding bigger and better grids.

To draw the grid designs more than once, students need to capture their designs in procedures. The idea of point procedures being tools is reinforced by writing other procedures using them. After drawing an arrow point by experimenting with positions, the arrow point is handier to redraw with a procedure.

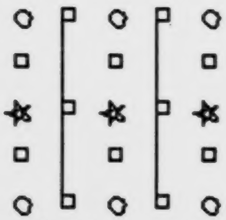
```
TO ARROW.POINT
A1 E5
A2 B3 C4 E5
C3 A2 E5
B1 C2 D3 E5
C3 B1 E5
END
```



Decorated Grids

The positional, or zip, procedures move the turtle to each point on the grid. However, drawings on the grid are not limited to straight lines. Small squares, circles and triangles can be added to the grid tools.

```
TO SQUARE
REPEAT 4 [ FD 5 RT 90 ]
END

TO CIRCLE
REPEAT 15 [ FD 1.4 RT 24 ]
END
```



These shapes are then added to the sequence of position procedures for decorated grids.

Tipps continued

Never Boring Geoboards

The geoboards are not a substitute for the time children spend with rubber bands and pegs. The sensorimotor learning with geoboards is just as essential as playing turtle is to their thinking process. Geologoboards are simply another way of expressing ideas about lines, shapes, positions, and colors and a new way of creating.

However, the geoboard and the coordinate geometry which underlies it are very different in nature from turtling and the intrinsic geometry which underlies it. Intrinsic geometry is based upon knowledge of where the turtle is here and now, not in relationship to other points. Some teachers may feel that coordinate geometry is superior because it is what is "taught" in schools. Intrinsic geometry is not inferior at all and has many strengths in describing events from an egocentric perspective. Logo has both geometric systems and opportunities to express ideas about each of them.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Teacher Feature

by

Rebecca Poplin

Featuring: Peter Rawitsch

As the holidays approach, many of us reflect on the most wonderful gifts (tangible or intangible) we have given or received. When we receive a special gift, with a childlike thrill, we want to share that gift with others. No gift could be more wonderful and intriguing to a user of Logo than that of new tools for thinking and creating. Peter Rawitsch, a first grade teacher in Glenmont, New York, has some delightful new tools that he has either created, found, or otherwise scrounged about for the money to purchase. Perhaps you would like to share them with others as Peter does.

Peter has discovered a fascinating construction kit, called *Capsela®* (Play-Jour, Inc., 200 Fifth Avenue, Room 1024, New York, NY 10010), which actually has gears inside clear plastic spheres. The spheres can be connected to each other with joining pieces to create land and water toys limited only by the imagination. Some of the spheres have speed reduction mechanisms. Wheels, propellers, and windmills can become a part of the construction. Talk about modularity!

Peter persuaded the PTA in his school to buy *Capsela* for his classroom. At first he used only the non-motorized pieces in his Microworld Center for off-computer activities. He was also careful not to put out pictures of construction projects so the children's creativity would not be limited. Even without motors, the constructions were dynamic and stimulated fascination with how one movement affects another.

Some *Capsela* models have "computer control" (via a microprocessor) and an infrared link. Peter is hoping to create an interface between Logo and the *Capsela* constructions during the year.

Peter has been interested in robots for quite a while. He uses the Big Trak, CompuRobot, and a Terrapin robot. His students predict the commands needed to move the robots and use the turtle to test hypotheses. Peter has written a grant for a Valiant Robot Turtle (available from Harvard Associates, 260 Beacon Street, Somerville, MA 02143, (617) 492-0660) for the teacher center in his area. Some of Peter's students are conducting a study to replicate animal movements with the robots.

Learning Logo

Peter Rawitsch's strong background in Early Childhood Education and the arts and his desire to develop the creativity of his students led him to Logo, though he does not have a strong interest in computers *per se*. He realized from studying a first draft of Terrapin Logo that the language puts the user in control of the computer. Peter taught himself to use Logo with the manual and without reading *Mindstorms*.

After attending a workshop on Logo in which the presenter claimed that Logo was not a good tool for group projects, Peter realized that teachers could benefit from his classroom experience by using Logo as an interdisciplinary tool, and developing both individual and group projects.

As Peter increased his work with other teachers, he began conducting inservice workshops and developing curricula, through which he has shared his experience and philosophy.

Rawitsch's Rules

One of Peter's gifts to teachers is what he calls *Rawitsch's Four Laws of Logo*.

1. **Logo is not easy.** It requires thinking, and thinking requires a lot of effort.
2. **Logo is not important.** The language itself is not important; what a child can *do* with it is important.
3. **Logo is not safe.** It is unpredictable. The best-laid teacher plans may go awry. A creative teacher must remain open to all kinds of opportunities and be ready to maximize the opportunities when they occur.
4. **Logo is not for everyone.** Many teachers cannot handle things that are not important and not safe. For those teachers, a different tool may be more appro-

Feature continued

priate. Some teachers get hung up in commands and miss the real objects of Logo. If we are going to use Logo, we need to agree that thinking skills and problem solving strategies are important.

Peter is quite interested in giving and receiving ideas about curriculum, robots, and other tools you may have found useful in your classroom. If you would like to communicate further with Peter Rawitsch, he would like to hear from you.

Peter Rawitsch
P.O. Box 254
Guilderland Center, NY 12085
(518)861-6637

If you know of creative teachers who are actively involved with using Logo and have special gifts to share through this column, please contact Rebecca Poplin, 2421 Fain Street, Wichita Falls, TX 76308.

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

NLXionary

A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"The Many Worlds of Logo," Part one of a three-part series on Logo, *Educational Computing* segment EC301, PBS weekly videotaped program, produced by Kentucky Educational Television (KET) in association with the EPIE Institute of New York.

>*Educational Computing* is a weekly half-hour TV series in magazine format for parents and educators. The programs typically cover educational software evaluations, comments from experts in the field of educational computing, and updates on computer news and technologies. Although this program is not a "reading," I feel it is worthy of comment.

>In this segment, Fred D'Ignazio reports on the variety of ways educators around the world are using Logo to address different educational needs and content areas. He finishes with a short interview of Papert.

>This is a nice overview of Logo's possibilities, especially for readers of the NLX, as opposed to a typical viewing audience who might need a simpler introduction to Logo and its philosophy. Some of the more unusual applications of Logo spotlighted were the treating of autistic children, creating weather graphics, teaching English as a second language to Japanese students, and using it as a mechanical design tool.

>The fact that many of those interviewed are university people involved in research left me with a mixed reaction. It confirms the "no ceiling" attribute of Logo as postulated by Papert, and, in that sense, excites and stimulates. It also confirms some feelings of inadequacy in that it appears to take considerable teaching talent and content area knowledge to help students go very far beyond Logo's introductory stages.

>Yet, that might be part of Logo's appeal for many of us - like the "diamonds are not dispersed on every street corner" philosophy of pursuing interests. Whatever the case, readers are encouraged to watch this series, particularly the next two segments, "Logo Under Fire," and "The Future of Logo."

(Ed. note: "The Many Worlds of Logo" (EC 302) is scheduled for rebroadcast on November 28 and 29. "Logo Under Fire" (EC 304) follows on December 26 and 27, and "The Future of Logo" (ED 308) on January 9 and 10. Local broadcast dates and times vary by as much as a week. Consult your local PBS station for more details, or call KET at (800) 354-9067.)

"Remembering the Path to Excellence," by Stanley Pogrow, *Phi Delta Kappan*, October, 1985.

>Four educators are asked to view education since 1985 from the fantasized vantage point of the year 2005. Pogrow's article is one of the four viewpoints, taking advantage of "hindsight."

>He speaks of the fizzle of microcomputers during the mid to late eighties as occurring because educators knew little about the learning process and the operation of schools, focusing instead on computer literacy, one computer in every classroom, Logo, and equal access to computers. Among other things, he says educators learned "that Logo didn't have any real effect on learning (though many children came to believe that Indians shot turtles at cowboys) ... With the wisdom of hindsight, it seems silly that substantial numbers of educators came to believe that kids could be made smarter spontaneously by moving a turtle around the screen."

>Pogrow maintains that the formula of infusing innovations into every classroom and subject according to a scope and sequence, regardless of whether or not teachers had mastered simpler teaching techniques, was an "ingrained bias of the time ..."

>His article goes on to cite exceptions to the above, and explains that the resolution of the problem came in the form of using interactive technologies to automate the routine teaching and learning of facts, "making it possible to concentrate more funding on specialists and master teachers, while providing a greater diversity of learning opportunities."

>Except for his sarcastic remarks about Logo, this is a very worthwhile and interesting article. His philosophy is quite similar to that of Marc Tucker, Director of the Carnegie Commission's Information Technology project. The US economy will be threatened unless schols begin graduating students who are able to work

NLXionary continued

in small cooperative groups turning innovative ideas into value-added products.

>I interviewed Pogrow while he conducted a workshop on his Higher Order Thinking Skills (HOTS) project, which is briefly described in the article. He acknowledged purposely using some inflammatory language as an antidote to Logo as religion. But, beyond that, the problems he has with Logo are more related to the structure of our current learning environments than with Logo itself.

>Logo is difficult for most teachers to relate to the traditional curriculum, he believes. It is not only difficult for students to generalize from Logo's problem solving approach to other areas, but even to have the opportunity for creating and then solving meaningful problems is rare.

>Lastly, Pogrow believes that Papert has underestimated the unwillingness of both teachers and students to deal with ambiguity. Problems and questions which are not open ended still rule the land. Although he doesn't believe in maintaining the status quo, he finds he can be more effective by combining good discussion and questioning techniques with software that is more transparent than Logo. (By transparent, he means requiring less effort to learn to operate the actual software, thereby allowing more effort to be spent on the learning process in the content area.)

>I don't have a particular problem with any of this. Many people have been geared up to expect the world of Logo, and have met with inevitable disappointment. If more teachers introduced their students to turtle graphics with the goal of having students acquire a deeper understanding and *feeling* for certain mathematical and geometrical properties and concepts, I suspect Pogrow would be less condemning.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Logo Disserts:

Dissertations Dealing with Logo

by Barbara Elias

Are there differences in the problem solving activities of gifted children and normal children who learn Logo? This question was examined in an exploratory study which involved sixteen children of ages 9 through 11. Differences were examined in the bugs and the debugging strategies of both groups. In addition, procedural nets (diagrams) were constructed from the projects they developed and examined for structure and level of complexity.

The children who participated in the study attended a public elementary school. Gifted children were selected using the school's identification procedure, Renzulli, Reis, and Smith's *Revolving Door Identification Model for the Gifted and Talented.*. An average group of the same age and sex was matched with the gifted group. None of the children had previous computer experience. An equal number of girls and boys was in each group.

Children worked in pairs with two researchers twice per week for ten weeks. After basic Logo concepts were developed, children were encouraged to explore and to "teach" the computer as they worked on individual projects. They were encouraged to sketch their projects first and discuss plans before beginning on the keyboard. As bugs were encountered, children were asked to describe the problems. Suggestions were provided for working through the debugging.

Data sources included dribble files, pencil drawings and notes, tape recordings, and teacher logs. A classification system protocol analysis was used to classify the problem solving activities. In such a system, activities are structurally sequenced into episodes and represented as procedural nets (diagrams). The system provided information on programming style, procedures, revisions, and bugs.

The researcher cautions that the results should be considered inconclusive because of the small number of children. The results obtained, however, indicate that differences were observed in the two groups. Gifted children developed more complex projects, used deeper hierarchial structures in projects, and wrote more procedures. Both groups had about the same number of bugs and no differences were noted in debugging style. Differences were noted in the frequency of bugs and in debugging actions. Children in the gifted group did more debugging after procedure writing, in contrast to the average children, who debugged more before procedures were written. Many individual differences were observed in the teams of each group.

Classroom teachers may find a problem solving model of the novice programmers in this study helpful for their own classroom observations.

>>>Maniatis, E. G. (1983) *An Analysis of the Differences in Problem Solving of Gifted and Non-Gifted Children Using the Logo Programming Language.* (Unpublished masters thesis, McGill University, Montreal, June 1983.)

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22903.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Listful Thinking

by Glen Bull and Paula Cochran

Instant Tools

Logo was designed to give users easy access to the power of a computer. As Papert has pointed out, the fact that children can use certain parts of Logo so easily has led to the mistaken impression that it is a children's language. Logo is also used in advanced applications such as college physics courses, and there is nothing inherently childish about the language. Nevertheless, many of us are particularly interested in the use of Logo with young children. In this column we will introduce a Logo tool called DO, which we hope will enable you and your students (even the young ones) to DO some exciting projects this month.

Troubles with Turtles

There are many ways to introduce Logo to young children. The difficulty of ensuring that it becomes empowering for them is a recurring topic in teacher training courses and in the Logo literature. Problems frequently reported by teachers include:

(1) the commands require too much typing for a young child
(2) the use of large, double-digit numbers does not match the child's developmental level

A form of Logo which is often called "Instant Logo" is one of the more popular solutions to these problems. In most Instant Logo programs (sometimes called "Easy Logo" or "Single-Key Logo"), the child only has to press a single key to make something happen on the screen. For example, the child might press F, immediately causing the turtle to move forward 10 steps.

Writing a short Instant program is not overly difficult, and, in fact, is a common exercise in a number of Logo texts. Usually these programs work by checking a single key against a list of possible commands. For example, if the child presses R, the program checks to see if the key pressed is F, B, etc., until it reaches R. Then it turns the turtle right 30 degrees. (Terrapin and Commodore Logo users omit the brackets in the procedure.)

```
TO INSTANT.CHECKLIST
MAKE "COMMAND RC
IF :COMMAND = "F [ FD 10 ]
IF :COMMAND = "B [ BK 10 ]
IF :COMMAND = "R [ RT 30 ]
IF :COMMAND = "L [ LT 30 ]
IF :COMMAND = "Q [ STOP ]
INSTANT.CHECKLIST
END
```

This traditional kind of INSTANT program has the virtue of simplicity. It's easy to see how the program works, and if you want to extend the program, you just keep adding more IF statements.

From a programming standpoint, this kind of structure is a common one, found not just in Logo, but in many other languages as well. One disadvantage of this structure is that, as you add more and more IF statements, the program takes longer and longer to check the list of possible commands.

It is possible to write a more elegant INSTANT.

```
TO INSTANT
DO RC
INSTANT
END
```

This Instant relies on a short procedure which will DO a command. In this case, each command must be defined as a one-letter procedure. Here is an example.

```
TO F
FD 50
END
```

DO is a short Logo procedure. It first checks to see if a procedure is defined. If the procedure is defined, DO runs the procedure. There are several possible uses for a tool like DO. This month we have chosen to concentrate on Instant applications. (See the end of the column for Terrapin and Commodore versions of the same procedure. If you use Apple Logo II and are typing in as you read, be sure to include the Q procedure below to stop your Instant.)

```
TO DO :COMMAND
IF NOT DEFINEDP :COMMAND [ STOP ]
RUN (LIST :COMMAND)
END
```

The DO command takes advantage of a sophisticated feature in Logo. The language treats data and programs in the same way. This lets us run a command directly, without searching through a look-up table of the kind used in INSTANT.CHECKLIST.

Another advantage of the new version of Instant is that it is easy for a teacher (or older students) to add new

Listful Thinking continued

commands. All that is necessary is to define new single key procedures. For example,

```
TO R
RT 30
END
```

This can be continued indefinitely, or at least until the alphabet is exhausted.

The Issue of Instant

One of the most highly touted characteristics of Logo is that it permits playful exploration of numbers. However, in traditional versions of Instant Logo, the child does not have an opportunity to input a number at all. This is just one of the limitations which has led critics to note that most versions of Instant take the power of Logo out of the child's hands (Papert, 1984; Tipps, 1983). This is true in many cases. Fortunately, an Instant Logo doesn't have to be so limited.

The version of INSTANT that uses the DO command can encourage construction of an Instant program which accommodates the child, but does not prevent access to Logo's capabilities. Our experience with looking at ways in which Logo can complement a pre-academic or early academic curriculum have lead us to require certain features in Instant programs we use.

Rule-of-Thumb Number 1

A written message on the screen should tell the child what is happening after each keypress. For example, when the child presses the letter F, we would like the word "FORWARD" to appear on the screen (the whole word, not the abbreviation). Since F is a procedure in the Instant we were working on above, it's a simple matter to edit it.

```
TO F
PRINT [ FORWARD ]
FD 10
END
```

Obviously, the developer of the Instant program has control over the screen message, which does not necessarily have to match the equivalent regular Logo command. Occasionally, for example, we have substituted the word WALK for FORWARD when working with cognitively delayed children (and yes, we have stickers all over the keyboard!).

Rule-of-Thumb Number 2

The child should have the opportunity to supply numeric inputs to procedures. How do we minimize typing and still provide a chance to explore with numbers? Keep your eye on that F procedure.

If you are using Terrapin or Commodore Logo,
a. Use PRINT1 in place of TYPE.
b. Use REQUEST in place of READLIST in the INPUT.NUMBER procedure.
c. Omit brackets around [OP INPUT.NUMBER].

```
TO F
(TYPE [ FORWARD ] SPACE)
MAKE "NUMBER INPUT.NUMBER
FD :NUMBER
END

TO SPACE
OP CHAR 32
END

TO INPUT.NUMBER
MAKE "NUMBER READLIST
IF :NUMBER = [ ] [ OP INPUT.NUMBER ]
OP FIRST :NUMBER
END
```

This revised version of our F command allows the child to type in a number. It also prints out the full word FORWARD, automatically followed by a space. Since the child may want to type more than a single digit (such as 10), the program waits for a carriage return before executing the Instant command.

Rule-of-Thumb Number 3

An Instant program should provide error messages that are meaningful to a child. In regular Logo, if you type FORWARD and press RETURN without entering a number, Logo is apt to present an error message such as:

```
FORWARD NEEDS MORE INPUTS
or
LOGO DOESN'T LIKE [ ] AS INPUT TO FORWARD
```

Most adults will recognize that Logo is telling them that by pressing RETURN they entered an empty list as input (right?), but this error message isn't helpful for a five-year-old child. It also isn't helpful if the program crashes when the user makes a slight mistake. We anticipate that there are two types of errors a young user of Instant will make:

(1) They will press RETURN without entering a number.
(2) They will type a letter instead of a number.

Our INPUT.NUMBER procedure should check for both of these possibilities, and provide a suitable error message.

Listful Thinking continued

```
TO INPUT.NUMBER
MAKE "NUMBER READLIST
TEST EMPTYP :NUMBER
IFTRUE [ PRINT [ PLEASE TYPE A NUMBER. ] ]
IFTRUE [ OP INPUT.NUMBER ]
TEST NUMBERP FIRST :NUMBER
IFFALSE [ PRINT SE :NUMBER [ IS NOT A NUMBER. ] ]
IFFALSE [ OP INPUT.NUMBER ]
OP FIRST :NUMBER
END
```

The Terrapin/Commodore version of this procedure is slightly different, and is shown below.

```
TO INPUT.NUMBER
MAKE "NUMBER REQUEST
TEST :NUMBER = [ ]
IFTRUE PRINT [ PLEASE TYPE A NUMBER. ]
IFTRUE OP INPUT.NUMBER
TEST NUMBER? FIRST :NUMBER
IFFALSE PRINT SE :NUMBER [ IS NOT A NUMBER. ]
IFFALSE OP INPUT.NUMBER
OP FIRST :NUMBER
END
```

The error message can be customized by the teacher. If "Please type a number" is not appropriate, it can be changed to any suitable message.

Rule-of-Thumb Number 4

An Instant program should encourage the use of numbers which are meaningful to the child. For a child who is just becoming familiar with small numbers, even the Instant described so far can be a bit frustrating. If she uses the numbers 1 to 10 as inputs, for example, not much is going to happen on the screen. In fact, small turns are frequently not visible at all in regular Logo (try it -- type RIGHT 2 and see what happens).

Dilemma: A preset number is extremely limiting, but use of the small numbers with which the child is familiar is unrewarding.

Solution: Make the small numbers visible, and thus more interesting. This involves multiplying the number typed by the child. We might decide that for every 1 step requested, the turtle moves 10 turtle steps, for a ratio of 1 to 10.

```
TO F
(TYPE [ FORWARD ] SPACE)
MAKE "NUMBER INPUT.NUMBER
FD :NUMBER * 10
END
```

The turn commands will need some adjusting, too. We found out the hard way, when one mystified six-year-old typed F(orward) 5 (which moved the turtle forward 50 turtle steps, in 5 easy-to-see jumps) and then R(ight) 5 (which barely turned it at all). The same range of numbers should be effective as input to distance and direction commands. We generally multiply our turn inputs by 10, so that an input of 3 results in a 30 degree turn, etc.

```
TO R
(TYPE [ RIGHT ] SPACE)
MAKE "NUMBER INPUT.NUMBER
RT :NUMBER * 10
END
```

If you are fairly new at Logo, we remind you to edit the F procedure above as a basis for the rest of the commands you need -- that's what we did to make R (don't type it all in again!).

Corollary to Rule Number 4

A young child should be able to see each step the turtle takes (and count along). While you are in regular Logo, type FORWARD 50 and press RETURN. How many steps did the turtle take? How do you know the turtle really took fifty steps? On most computers, it looks as though the turtle took one giant step instantaneously (pun intended).

The turtle should pause in between steps. Then the child will be able to count along as the turtle moves. We often add a WAIT procedure, causing the turtle to stop briefly after each step. If your students count more slowly, just increase the length of the delay. For example, change WAIT 20 to WAIT 40. (Terrapin and Commodore users will find their version of F and WAIT at the end of the column. WAIT is a built-in or primitive procedure in Apple and IBM Logo.)

```
TO F
(TYPE [ FORWARD ] SPACE)
MAKE "NUMBER INPUT.NUMBER
REPEAT :NUMBER [ FD 10 WAIT 20 ]
END
```

DO Stop!

We interrupt this program to bring you a helpful hint. If you are going to make an INSTANT program that uses the DO command, there is one procedure that you certainly do not want to omit. Be sure to include a command to stop the program! The procedure that follows will allow you to exit Instant, by going to the top level of Logo. (Terrapin and Commodore version at the end of the column.)

```
TO Q
PRINT [ QUIT INSTANT ]
THROW "TOPLEVEL
END
```

Listful Thinking continued

You will also probably want to add to your Instant the capability to CLEARSCREEN, go HOME, and put the pen UP and DOWN.

Customizing Instant

Use of Instant Logo can be empowering for even young children. Instant can be used to facilitate a wide range of instructional objectives. We have suggested that an effective Instant Logo program should have the following characteristics:

1. appropriate turtle step length
2. appropriate turtle step speed
3. visible results with small numbers (for young children)
4. messages telling what is happening on the screen
5. meaningful error messages
6. easy exit and resistance to crashing

We have explained just a few of the modifications to traditional Instant that we use and that we think are important if young children are going to get the most from early exposure to Logo. If this is a topic of special interest to you, get in touch and we will gladly share more of our ideas along this line.

Instant Measures Up

Is a well-constructed Instant program useful for anything besides early turtling experiences? Can older students also use Instant to achieve content-related objectives? We think so. For example, what if we make each Instant turtle step equal one inch (a ratio of about 1 : 32)? The procedure for making the turtle move forward would look like this: (Terrapin users change TYPE to PRINT1.)

```
TO F
(TYPE [ FORWARD ] SPACE )
MAKE "NUMBER INPUT.NUMBER
REPEAT :NUMBER [ FD 32 SLASH WAIT 20 ]
END

TO SLASH
RT 90 FD 2
BK 2 LT 90
END
```

Now the command F(orward) 6 results in the turtle moving forward six inches. SLASH adds a little marker after each step (each inch). Suddenly we have a new way to explore estimation and measurement. Picture in your mind's eye ...

Mike and Joe are doing their math worksheet. They each have a job in this activity. They choose who does what, and they are allowed to change jobs. At first, Mike will be the Estimator, and Joe will be the Measurer. They are sitting at a computer with a plastic overlay taped to the monitor. They have a worksheet, a pencil, and a ruler (or tape measure).

During the next twenty minutes or so, Mike and Joe are going to estimate and then measure distances between 2 and 10 inches. The worksheet asks them to find out how far it is between the buildings on the overlay. For example, how far is it in inches from the red house to the blue house?

	1st Estimate (by eye)	2nd Estimate (by turtle)	Actual Measurement (by ruler)
Red house to blue house:			
School to grocery:			

The boys are supposed to estimate twice before they actually measure the distance. Mike and Joe make a guess. They think it's about 4 inches from the red house to the blue house. That's the first estimate. Mike moves the turtle to the red house, and then moves F(orward) 4. Oops! The turtle is still about an inch from the red house. They estimate that the real distance is close to 5 inches. Now Joe checks their work with a ruler, and they fill out their worksheet.

The next day, they have a similar assignment, except now one turtle step equals a centimeter. Oh! How big is a centimeter, anyhow? (A ratio in normal turtle steps of about 1 : 12.) By estimating, moving the turtle, and perhaps checking themselves with a ruler, they can find out. They can also pose measurement tasks (in turtle steps) for each other, now that they have the idea. How far is it to the grocery from the red house, if you go to the blue house first? What is the shortest distance between two houses? Is it farther in inches or centimeters? (Neither-- HO HO HO).

We hope that you and your students DO enjoy the holiday season. Best wishes for Instant good cheer!

References

- Chmielowski, C. M. "Logo Trail of Tears."
  National Logo Exchange, November 1984, pp. 1-2.
- Lough, T. "Slow Turtle Moves Clearly."
  National Logo Exchange, September 1983, pp. 7-8.
- Papert, S. "Misconceptions about Logo."
  Creative Computing, November 1984, pp. 229, 232.
- Tipps, S. "Tipps for Teachers: The issue of Instant."
  National Logo Exchange, December 1983, pp. 6-10.

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a doctoral student at the University of Virginia. She is interested in Logo applications in language arts and speech-language pathology.

Listful Thinking continued

Here are the procedures for users of Terrapin and Commodore Logo:

```
TO DO :COMMAND
IF NOT DEFINED? :COMMAND STOP
RUN ( LIST :COMMAND )
END

TO DEFINED? :PROCEDURE
OP NOT TEXT :PROCEDURE = [ ]
END

TO INPUT.NUMBER
MAKE "NUMBER REQUEST
TEST :NUMBER = [ ]
IFTRUE PRINT [ PLEASE TYPE A NUMBER. ]
IFTRUE OP INPUT.NUMBER
TEST NUMBER? FIRST :NUMBER
IFFALSE PRINT SE :NUMBER [ IS NOT A NUMBER. ]
IFFALSE OP INPUT.NUMBER
OP FIRST :NUMBER
END

TO Q
PRINT [ QUIT INSTANT ]
TOPLEVEL
END

TO F
(PRINT1 [FORWARD] SPACE )
MAKE "NUMBER INPUT.NUMBER
REPEAT :NUMBER [ FD 10 WAIT 20 ]
END

TO WAIT :NUMBER
REPEAT :NUMBER [ ]
END
```

TurtleTips

by Jane Toth
and Donna Lanyi

We've been hearing a lot lately about incorporating Logo activities into the established curriculum. Many teachers said that they didn't have the instructional time to add a new subject, such as Logo, into their already hectic schedule. We readily agreed with these comments. But if we want to use Logo in the content areas, how do we begin? To be frank, we were stumped for a while, until we started to analyze each subject in terms of objectives. The light bulb finally came on. Here are some ideas for using Logo to enhance areas such as social studies, language, math, and the like.

The Turtle Teacher

Our students love a challenge. The majority of them enjoy the thrill of competition. We chose the topic of robots. We had been talking about robotics as part of our computer literacy program. As a class, we brainstormed about what marvelous features we'd each like our own personal robot to possess.

Of course, the students came up with some super ideas and a few items which would make their lives easier -- a homework completer, a room straightener, a chore completer. Why, what else would you want a robot for??

The children loved adding unique features to their robots. They then spent several days drawing their robots with Logo on the computer. We printed out their robots and used them for a bulletin board display.

In our social studies series, the fourth grades study US geography. A large portion of the book is devoted to map skills. In last year's December column, we talked about having the children work with mazes. A natural extension of this is to have our students draw their own Logo maps. Steve Tipps' column in last month's NLX contains some excellent ideas for this.

First, we had the class work on maps of the classroom. We had them plan it out, walk it out, and design it on the computer. They loved this activity. After fairly successful classroom maps, we took it a step further and had the students map out their routes to school. This helped to open their eyes and be keen observers on their way to and from school. You could extend this idea even further by measuring (or estimating) distances and drawing *scaled* maps with Glen and Paula's Instant from this month's column. (Ed. note: See "Listful Thinking" elsewhere in this issue.)

One of our recent reading stories was about the Clown College in Venice, Florida. Every year, we've had our students design their own clown faces on paper. This year, we utilized the computer. The students loved adding colorful facial features, a few tear drops, or some wild hair. Each clown designed had its own unique personality. The students then took their clowns and used them in a creative writing story as the main character. Doris Schroeder's Page 1 idea for a LogoShow is a perfect complementary activity for projects such as this!

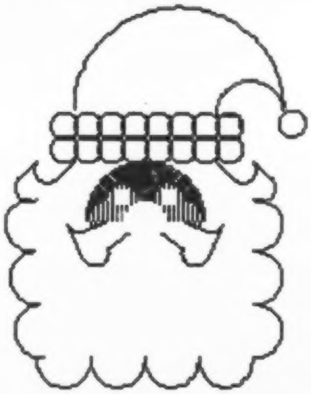
We have finally begun integrating Logo into our existing curriculum! No one said it would be easy, but it *is* both fun and exciting!

Procedure of the Month

Students love to share products they have completed. Use the Santa Claus procedure to get your class involved in creating festive holiday ideas. Students could use their Logo creations as Christmas cards, wrapping paper, or even ornaments. How about collecting students' holiday pictures and combining them into a Christmas coloring book? The Santa Claus

TurtleTips continued

picture below was created by Pat Arnold, a first grade teacher in Wooster, Ohio. If you are interested in a listing of the procedures for the drawing, please send us a self-addressed stamped envelope.



We invite any teachers or elementary students to send Logo procedures or ideas to be shared in this NLX column. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher to Teacher
A Review of New Logo Materials

by James Fry

This month's column features a guest review of a publication for all Atari Logo users written by Sharon Yuille, a teacher in Flint Michigan, and comments on a set of programming manuals for Apple Logo.

Atari Logo Users

Logo Aide, by Barry A. Hoglund, 1984, published by August Publications, P.O. Box 67, San Rafael, Ca 94915. $29.95.

Logo Aide is exactly that - an aide that anyone who teaches Logo should be able to call upon. Barry Hoglund speaks in a language that is sophisticated enough to be interesting to adults but at the same time simple enough so that much of the material could be used straightforwardly with students. Mr. Hoglund also treats the multiple turtle (there are four) feature of Atari Logo with the importance it deserves.

Target audience: all grades

Content: The book is organized into seven sequential phases, beginning with simple turtle graphics and progressing through multiple turtles, writing procedures, editing shapes, using variables, and two forms of animation. Each phase has a teacher's guide, with step by step instructions for introducing the primitives for each section. There are also many diagrams and teaching aids.

Format: The book is well-constructed, durable and easy to work with. Commands are printed in large letters on separate sheets so that a teacher could pull them right out of the book and post them by the computer. The book (notebook style) is printed on heavy paper and is divided into 7 sections or phases. Each phase begins with a guide that is very clear and easy to understand. Each section includes activities to do and items to discuss before working on the computer.

Strengths: The Multiple Turtles section was not only clear and concise, but fun to do and appeared to be easy to teach. It seems to be a good idea for the kids to have access to all four turtles before moving into the concept of coordinates. The shape editor section is ample with plenty of exercises and designs.

The section on variables is invaluable. It seems so often that students and teachers do well until they get to variables; then they get frustrated and sometimes drop back. Mr. Hoglund has made learning variables easy to understand and exciting enough so that readers would want to add more variables to their procedures.

Record keeping sheets were excellent. The placement activities would be very helpful to a teacher. I especially liked all of the activities for little ones. My favorite provided a youngster with a reproducible turtle complete with a chalk "pen" for chalkboard precomputer activities.

Suggestions: Although Mr. Hoglund includes many exercises in the procedures section, I wish he would have included more samples of simple procedures using block letters. I also suggest that more practice sheets and simple geometric figures be added before moving into rotating designs.

I have a concern about teaching color before mastering geometric designs, but I plan to try it this way before passing judgment. However, the book is set up so that a teacher or parent could rearrange parts of a section depending on the group or child's needs.

This is an excellent resource book, reasonably priced. Perhaps best of all, Mr. Hoglund encourages teachers to reproduce pages from the book for the classroom.

-Sharon Yuille, guest reviewer, Flint Community Schools, Flint, Michigan

Teacher to Teacher continued

Apple Logo Users

Apple Logo Programming Primer, by Donald Martin, Marijane Paulsen, and Stephen Prata, 1984, published by Howard W. Sams & Co., Inc., 4300 West 62nd St., Indianapolis, Indiana, 46268. $19.95

Target audience: older or experienced Logo students

Content: This book is designed to teach Logo as a structured programming language and be a guide for top-down structured program design. It consists of fifteen chapters starting with an introduction to turtle graphics for a beginner to advanced list processing chapters for an advanced Logo programmer. The book also contains an excellent appendix.

Strengths: The layout of the book is well done and easy to follow. The authors have used what they call the "box chart" method to show Logo procedures and to make it easier to follow and understand. The use of color in the text to highlight areas also helps to clarify ideas.

There are many example procedures or "Logo programs" that can be keyed in and saved for future use in some of your own Logo programs, or for reference. The appendix section containing a summary of Logo commands is excellent and would be a valuable aid to anyone working with Apple Logo. The other appendices on ASCII codes (with chart), managing workspace, error messages, startup, and Logo features are also excellent and valuable.

Companion: There is also a companion book, *88 Apple Logo Programs*, by the same authors and publisher, that goes along with the *Primer*. It gives many more procedures and programs that can be used as is or combined with others of your own. For us who are lazy typists, the 88 programs come on a disk that is available with the book.

Do you know of any new Logo materials and books? Did you find them useful in teaching? If so, we invite you to send information to the Logo Information For Teachers (L.I.F.T) group, PO Box 5396, Plymouth, MI 48170, or send your comments on CompuServe to 76317, 565.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI and is a co-founder of the Logo Information For Teachers group.

Q and A

by Jim McCauley

Q. I am trying to enter a very long list as data for a Logo program, but I'm having a problem. I can type in about forty numbers, but then the computer just beeps at me and won't let me type in any more. What's wrong?

A. Nothing's wrong. You have hit an interesting limit in the design of most Logos: the text input buffer. As you probably know, when you type things into the Logo editor, you can type in lines that are pretty much as long as you wish. However, if you are typing into Logo from outside the editor, there is usually a limit to the number of characters you can type before pressing the RETURN key. For most versions, the limit is 128 characters.

This limit exists for pretty good reasons. First, the architecture of most computers assumes that information typed directly into a running program (like a program you have written in Logo , or the Logo interpreter itself) is going to be limited in length. There is usually a part of the computer's memory that is set aside especially for this purpose; it is commonly called the text input buffer. The availability of the buffer makes the programmer's job of designing an input system much easier. The Logo editor and many other programs (word processors, for example) allow very long lines to be entered because the programmers have used special techniques to get around the size limit of the text input buffer.

If you wish to associate a very long list of data wih a variable, here are two ways to get the job done:

If you want to assign the list to a global variable with MAKE or NAME at the time that you are writing the program, you should use the "name editor." The name editor is actually just a variation on the procedure editor; it works in exactly the same way, but you call it into play with the command EDNS (or, EDIT NAMES; the command varies slightly between implementations). You will be presented with an editing screen full of all of the global values that you (or your programs) have initiated with MAKE or NAME, and you can edit them in exactly the same way that you edit procedures.

If you want to create a very long list while a program is actually running, you must use a different method. This pair of procedures will give you an idea of how this works:

```
TO INPUT :PROMPT
PRINT :PROMPT
PRINT [ Press RETURN at this prompt when finished. ]
OUTPUT DO.INPUT READLIST [ ]
END

TO DO.INPUT :INP :LST
IF :LST = [ ] [ OUTPUT :LST ]
PRINT :PROMPT
OUTPUT DO.INPUT READLIST LPUT FIRST :INP :LST
END
```

Q and A continued

The idea here is to build up a very long list one element at a time, then output the whole thing once it's built. In most Logos, you can input very large lists in this manner.

Type in these procedures, and put them to work with something like:

```
SHOW INPUT [Please type in a number.]
```

and see how big a list your Logo allows. It might be as long as several hundred items for the larger Logos.

I hope this gives you some ideas about the assembly and manipulation of long lists in Logo.

Each month, I will answer selected questions in this column about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to: Jim McCauley, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a self-addressed stamped envelope. You may also send questions to me on CompuServe at 70014,1136.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

NLXual Challenges

by Robs Muir

Making List Operators

During the past 8 months, the friends of these NLXual Challenges have been exploring around a fundamental issue in mathematics and logic. These explorations have taken form in challenges asking us to write an IF procedure (March 1985), and a "user-defined" FORWARD procedure (May 1985).

In true Logo fashion, we have been having so much fun with these problems that some of us didn't realize that we were exploring profound, fundamental ideas -- ideas that link the world of computation to the disciplines of mathematics and formal logic.

BUTFIRST

Before we get into this month's challenge, let me share an observation that E. Paul Goldenberg made: When introducing Logo's capacity to manipulate words and lists, most books, articles, and classes typically teach first the primitives FIRST, LAST, BUTFIRST, BUTLAST, and SENTENCE. Occasionally, WORD is included. The standard sequence usually progresses to ITEM, COUNT, and the Logo predicates EMPTYP (EMPTY?), LISTP (LIST?), and WORDP (WORD?). Paul states, "I am not sure I have ever seen LIST, FPUT, or LPUT taught except for the sake of completeness."

Instructors often make distinctions between operations that put lists (or words) together and take them apart. (Ed. note: See Jim McCauley's "Q and A" in the October 1984 NLX.) How often is the related issue of reversibility raised? If we dissect a list in a particular way, how can we reliably reassemble the parts into our original list? Here is an example.

```
TO DONT.CHANGE.MY :LIST
OP____________(FIRST :LIST)(BF :LIST)
END
```

What Logo primitive should you put in the blank so that this procedure will return an object that is identical to its input whenever the input is a non-empty list? Stop reading right now and solve this problem! (Hint: the answer is *not* SENTENCE.)

It is nice to be able to put back together what one has taken apart, not only for the sake of mathematical logic, but also for purely practical reasons. It is easier to describe the undoing of the FIRST / BUTFIRST pair than it is to describe exactly what SENTENCE does. It is also not any harder to learn the inverse of FIRST / BUTFIRST that it is to learn SENTENCE.

OK ... since you've successfully filled in the blank, we can tell *you* the answer. That inverse is FPUT. FIRST and BUTFIRST provide a way of dissecting a list, element by element. If you want to access the second element of a list, all you need to do is take the FIRST element of the BUTFIRST of the list. A similar approach gets you deeper into the list.

FPUT allows you not only to reconstruct what you have torn asunder, it also permits you to assemble other structures from the building blocks you may collect. For example, the procedure

```
TO BUTSECOND :LIST
OUTPUT FPUT (FIRST :LIST)(BUTFIRST BUTFIRST :LIST)
END
```

will output all but the *second* element of its input list (provided, of course, that the input list contains a second element).

Taking Paul's observations and our past record of torturing and crippling our poor Logo interpreter , let's now suggest a challenge or two for the coming month.

NLXual Challenges continued

Here They Come

Challenge #1 is to build equivalents of LAST, BUTLAST, LIST, SENTENCE, and LPUT using only FIRST, BUTFIRST, FPUT, and WORD for dissection and reassembly. You may use any other Logo primitives you'd like, but Paul suggests you think small. It is possible to do everything with a very limited vocabulary. (You can get along quite well without COUNT, MEMBERP, ITEM, and EMPTYP, for example, though you might decide it is convenient to build one or more of these for use in your solution.) Don't bother dealing with parenthesized versions of LIST, SENTENCE, and FPUT, but rather assume that each takes exactly two inputs always. Do insure that your procedures (you might call them MY.LIST, MY.SE, etc.) perform all of their tasks exactly as their primitive versions do.

Challenge #2 is to figure out the best building blocks from which to begin. We suggested that you use FIRST, BUTFIRST, LPUT, and WORD. Perhaps another starting set might make the task easier? Perhaps any set that includes some dissection equipment (such as FIRST) and some recombiners (like SENTENCE) allows you to build all the others? Or perhaps some tools are truly more primitive than others? What is the smallest set you can use?

Some workers in computer science, as well as workers in formal logic and mathematics (notably A. Church's work with *lambda calculus*), have dealt with this challenge in one form or another. For example, P. J. Landin's work, "The Mechanical Evaluation of Expressions," (*Computer Journal*, 1964) proposes that seven objects (null, h, t, nullist, prefix, if, and Y) provide a basis for describing a wide range of functions. Can you guess which seven Logo primitives correspond to Landin's?

And you thought Logo was only good for drawing pictures...

Send your solutions, along with a stamped self-addressed envelope, to NLXual Challenges, Attn: Robs Muir, 1688 Denver Avenue, Claremont, CA 91711. We will compile several interesting solutions and redistribute them to all contributors. By the way, you may send solutions to any past Challenges at any time.

Robs Muir is a teacher in the Claremont (CA) Unified School District and Claremont Graduate School

Logo Notices

So far, no one has come up with a "fix" for the Imagewriter overprinting with Apple Logo II (see September 1985 NLX), although solutions are now available for Terrapin and Apple Logo (see October NLX). Anyone figuring this one out please send us a solution which can be shared with NLX readers. If you like such technical challenges, send $3 to Tom Weishaar, 10026 Roe Avenue, Overland Park, KS 66207, and ask for a copy of the November issue of his outstanding *Open-Apple* newsletter. The lead article in that particular issue is devoted to solving printer problems, and may contain a clue to this dilemma. (By the way, you might want to subscribe to his newsletter - it's a real deal at $24 per year.)

Regina Sapona, editor of the NLX's popular *Logo Research Directory*, is preparing for an updated edition. If you would like to be listed, send a summary of your research activities and interests to Regina Sapona, 2715 East Towers Drive, #11, Cincinnati, OH 45238. For a copy of the current directory, send $3 to NLX Research Directory, Box 5341, Charlototesville, VA 22905.

Only one month remains before the International Logo Exchange newsletter begins publication! The first reports from the continental Logo field editors are in production and contain news about exciting Logo activities from all around the world! If an international Logo perspective is important to you, subscribe now, and begin with the first issue in January. The newsletter is published bimonthly (6 issues per year), and costs US$19.50 for addresses in the USA, Canada, and Mexico, and US$24.50 elsewhere. The International Logo Exchange, PO Box 5686, Charlottesville, VA 22905.

Kentucky Educational Television announces the rebroadcast of the "Educational Computing" television program series which features Logo. Here is a schedule of the rebroadcast dates. Consult your local public television station for *local* airdates and times. For more information, write to: Mary Campbell, Kentucky Educational Television, 600 Cooper Drive, Lexington, KY 40502, or call (800) 354-9067.

Nov 28 & 29	"The Many Worlds of Logo"
Dec 26 & 27	"Logo Under Fire"
Jan 9 & 10	"The Future of Logo"
Jan 23 & 24	"Teacher Training"

The National Logo Exchange, copyright © 1985 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717.

Editor...Tom Lough

THE NATIONAL LOGO EXCHANGE

The Logo Newsletter for Teachers

Volume 4 Number 5 FORWARD 100! January 1986 Page 1

Sometimes Logo Teachers Should Teach

by Marge Stouffer

"That's not what I wanted the turtle to draw. There's something wrong with my procedure!"

"Well, why not change it? You can use the editing commands. Just type EDIT and the name of your procedure."

"Like this?"

"Don't forget the quotation mark."

"Now what do I do? I see the line I want to change. How do I erase that line and put something in its place?"

My fifth grade students are eager Logo users. They enjoy the experience of experimenting. But, I found they become frustrated when they have to "discover" things like Logo syntax or how to use the Logo editor. They have difficulty making corrections and changing their procedures. They get discouraged and sometimes give up.

The philosophy of Logo includes, "It's OK to make mistakes." Students are invited to experiment, to make errors, to change procedures, and to correct their mistakes in order to nurture their imagination and originality. All of this requires a working knowledge of the editing process.

I believe that working with Logo increases students' creativity. I have seen it help develop individuality and foster logical thinking skills. It gives my students the opportunity to experiment. But, until my students learned how to use the editing tools to make their changes, they were not able to edit their work effectively.

Some teachers suggest giving students a copy of the editing commands and letting them "discover" how to use them. However, I have found it more effective to *teach* these specific skills and how to use this specific Logo tool.

Giving such direct Logo instruction does not provide for a lot of experimentation at first. But it does help students to experiment more freely later, since the frustrations of having to "discover" the proper use of the editing tools will not get in their way.

By teaching this part of Logo instead of asking the students to "discover" it on their own, I believe my students have become much more open to making changes and corrections than they would have been.

I would like to share a 16-step Logo editing exercise I have developed for use with my fifth graders. It has been effective in helping them learn how to use the various capabilities of the Logo editor, and has helped to avoid many confusing and frustrating situations which arose in the past.

When I see that a number of my students are ready for editing, I set aside a period of instruction specifically for teaching how to use the Logo editor. After distributing a handout to the students, I ask them to complete each of the 16 short activities one at a time. I keep the class together this first time through, making sure that everyone's screen looks as it should. When the class has completed the set of activities, I work with smaller groups of students who may need help on particular instructions or concepts. Sometimes I need to reteach the entire set of activities.

Logo Teachers continued

The students who are ready for the editing skills begin using them immediately. Others postpone using them until the appropriate time.

Here are my 16 activities for Apple Logo. If you use them with your students, you might want to add additional keys or change them to suit your grade level.

1. Type a simple procedure:

```
TO SQUARE
REPEAT 4 [ FD 30 RT 90 ]
END
```

2. Type EDIT "SQUARE

3. Make the cursor move to the second line of the procedure by typing CTRL-N.

4. Make the cursor move to the end of this line by typing CTRL-E.

5. Erase this line by using the left arrow.

6. Type: REPEAT 3 [FD 30 RT 120]

7. Type CTRL-C to define the new procedure. Type SQUARE and watch the screen.

8. Type EDIT "SQUARE.

9. Move the cursor to the line after END by typing CTRL-N.

10. Move the cursor back to the second line by typing CTRL-P.

11. Make the cursor move to the beginning of this line by typing CTRL-A.

12. Type: REPEAT 6 [FD 20 RT 60]

13. Make the cursor move to the end of this line by typing CTRL-E.

14. Erase the extra words by using the left arrow.

15. Type CTRL-C to define the procedure. Type SQUARE and watch the screen.

16. Practice changing the SQUARE procedure in other ways using CTRL-N, CTRL-E, CTRL-A, CTRL-P, and the left arrow key.

Marge Stouffer has taught elementary grades for the Palos Verdes (CA) Unified School district for 19 years, and has used Logo with her 5th grade math class for 3 years.

From The Editor

by Tom Lough

"Why have you remained so enthusiastic about Logo for so long? Don't you ever get tired of it?"

I had asked for questions from the persons attending my session at a recent educational computing conference. I expected and received the usual questions dealing with methods of teaching, teacher training, and capabilities of specific Logo versions. But here was a question of a different sort!

I hesitated for a moment, not because I had not thought about the question before, or that I did not have a ready answer. During that pause, my mind went back to a particular meeting of a human development course I was taking at the University of Virginia.

The instructor had introduced an idea for discussion: "Why is it that we can purchase a painting, hang it in the foyer of our home, and then move it to the attic a few months later, while we purchase another, hang it, and never tire of looking at it?" After several class members had discussed their responses, the instructor suggested: "For me, it seems to be because I never reach closure on the message of the second painting."

At the time I thought that was an interesting idea. Never reaching closure on the message means there is always something else to think about. That is how it is with Logo for me! Before addressing the question on the floor, I paused to savor that idea once more. Popping back to the present, I tried to convey this feeling somehow in my response.

Yes, my interest in Logo continues in large part because I "never reach closure on the message." No matter what I do, there is always another aspect to explore or another enhancement to consider. I have never written a program which is "complete." Every day, I think up new projects to explore with Logo. And just when I feel that I am getting close to understanding a concept (such as recursion), along comes a new idea forcing me to rethink everything I thought I "knew."

In addition to a personal dimension, my response also has a professional dimension. I never tire of Logo because of what I see it doing for students!

When taught in an accepting environment with the opportunity for student ownership of project work, Logo can help shape a more healthy attitude toward learning. With this attitude, students will never reach closure on the message of Logo ... that it is a language for lifelong learning.

My New Year's wish is that you, too, never reach closure on Logo's message!

FD 100!

Tom

NLXtra
A Supplement to the National Logo Exchange Newsletter

Volume 4 Number 5 FORWARD 100! January 1986 Page 1

West Coast Logo Conference Enjoyed by All

by Parnelli Sharp
and Gail Lovely

You should have been there!

Culver City, California, was the site of the first West Coast Logo Conference, held on November 21 - 23, 1985. Sponsored by Pepperdine University, International Council for Computers in Education (ICCE), Computer Using Educators (CUE), and the Los Angeles County Teacher Education and Computer Center, the three day event drew a crowd of over 900 attendees. A variety of Logo activities were offered, from beginning turtle graphics to advanced list processing.

Comments from the participants ranged from "Fabulous!" to "When is the next one?" Everyone was enthusiastic and seeking ideas about Logo use in the schools at all levels.

The conference consisted of concurrent events from which attendees selected activities to suit their needs. Such activities included demonstrations from teachers using Logo in their classrooms, hands-on workshops, vendor displays of the most current Logo materials and resources, Logo leaders speaking on Logo topics, panel discussions focusing on a variety of Logo issues, sessions in which you could chat informally with well known Logo workers, and presentations from such noted people as John Allen, David Thornburg, and Seymour Papert.

As one of the main after-dinner speakers, David Thornburg realized that his job was to inform and to entertain. He succeeded supremely at both! If you have not yet heard his "praises" of BASIC, you are missing a fantastic talk.

Some feel that the highlight of the conference was Prof. Papert's keynote address. He demonstrated (and debugged!) the latest development from Logo Computer Systems, Inc. ... a combination of Logo and a programmable word processor! Following his address, he answered a variety of questions from the audience. One of his most enlightening responses was to the question, "What advice do you have for a teacher just starting out with Logo?" He answered without any hesitation, "Just have faith in the kids!"

Conference proceedings are to be published by the ICCE. Details will be forthcoming.

Will there be a second West Coast Logo Conference? No one can say for sure right now. Stay tuned to the NLX for further information.

Parnelli Sharp is a kindergarten teacher in the Santa Monica (CA) Unified School District, and a mentor teacher in Logo. Gail Lovely teaches second grade in the Los Angeles (CA) Unified School District, is a mentor teacher in computers, and a consultant for Walt Disney Educational Media Company. Both Parnelli and Gail served as coordinators for the West Coast Logo Conference.

International Logo Exchange Newsletter Begins Publication

The National Logo Exchange extends a warm welsome its new sister publication, *The International Logo Exchange* (ILX) newsletter! With its January 1986 primiere issue, the ILX begins a bimonthly publication schedule to bring Logo news from an international perspective to readers around the world. ILX editor Dennis Harper has assembled an outstanding staff of field editors on each continent to provide timely reports on Logo activity in their area.

NLX readers everywhere are invited to provide their field editors with information about their Logo work. Write to the field editors at the following addresses.

Africa: Fatimata Seye Sylla, Laboratoire Informatique et Education, Ecole Normale Superieure, BP 5036, Dakar, Senegal, West Africa.

Asia: Hillel Weintraub, Doshisha International High School, Tatara, Tanabe-cho, Tsuzuki-gun, Kyoto-fu, 610-03 Japan.

ILX continued

Australia: Anne McDougall, Faculty of Education, Monash University, Clayton, Victoria 3168, Australia.

Europe: Richard Noss et al., London University Logo Group, University of London Institute of Education, Department of Mathematics, Statistics, and Computing, 20 Bedford Way, London, England WC1 HOAL.

Latin America: Horacio Reggini, Av. Alem 1074, 1001 Buenos Aires, Argentina, South America.

North America: Michael Friendly, York University, Psychology Department, Room 210 BSB, Downsview, Ontario M3J 1P3 Canada. Canadian and USA NLX readers are especially encouraged to send information about Logo activities on a regular basis to Michael.

Subscriptions to the ILX are available for US$19.50 per year for addresses in the USA, Canada, and Mexico, and for US$24.50 per year elsewhere. Send subscription orders to ILX, PO Box 5686, Charlottesville, VA 22905.

1986 ILX Tour to Iceland and Netherlands Announced

The International Logo Exchange (ILX) announces its 1986 Iceland / Netherlands Logo Tour. Led by ILX editor Dennis Harper, the 14-day tour will include Logo workshops, presentations, school site visits, lectures, dinners with host country Logo educators, and many other exciting Logo activities. Logo educators at all levels are invited to participate in this international Logo event.

The tour will visit both Iceland and the Netherlands, spending approximately one week in each country. All tour participants will deliver one workshop or paper to Icelandic or Dutch colleagues.

The tour departs New York on August 11, 1986, and returns on August 25. (Departures from other cities can be arranged easily.) The $1589 price encompasses all air fares, transfers, tours, hotels, 2 meals per day, and all educational activities, including three quarter units of graduate credit through the University of California.

For more information on this exciting tour, write to Educational Spectrum, Inc., 188 Summerfield Street, Scarsdale, NY 10583, or call (914) 723-2242 (collect, if you wish).

Logo Notes

Horacio Reggini's latest book, *Ideas y Formas*, contains a set of procedures recommended for anyone who uses Logo. Reggini's procedures permit one to draw with Logo in three dimensions, and then manipulate the drawing about any axis! Although the text of the book is in Spanish, the procedures are written in LCSI style Logo in English. For more information, write to Emece Distribuidora, Alsina 2062, Buenos Aires 1090, Argentina.

Professor Seymour Papert is collecting anecdotal information on Logo class happenings. He invites NLX readers to send him informal reports of interesting interactions observed in classroom settings and elsewhere as students are working with Logo. Write to: Prof. Seymour Papert, MIT E15-309, 20 Ames Street, Cambridge, MA 02139.

Tuttle Products announces Logo templates for the Apple and Commodore 64 computers. These colorful laminated overlays contain many Logo commands, with examples and diagrams. Templates cost $5.95 each, plus $1.50 for shipping and handling, and are available for Apple Logo, Krell and Terrapin Logo, and Commodore Logo. Tuttle Products, Inc., PC Box 26981, Tamarac, FL 33320-6981.

Logo Computer Systems Incorporated (LCSI) has verified a bug report on Apple Logo II. According to Ian Garmaise, LCSI Microsystems Manager, there is a small range of values returned by POS which are not useable by DOT. This is apparently due to the method used to calculate POS in conjunction with the current SETSCRUNCH. Ian sends the following range-adjusting procedure.

```
TO CORFIX :INLIST
IF (FIRST :INLIST) > 139 [MAKE "INLIST FPUT 139 BF :INLIST]
IF (FIRST :INLIST) < -140 [MAKE "INLUST FPUT -140 BF :INLIST]
IF (LAST :INLIST) > 119 [MAKE "INLIST LPUT 119 BL :INLIST]
IF (LAST :INLIST) < -120 [MAKE "INLIST LPUT -120 BL :INLIST]
OUTPUT :INLIST
END
```

Here is an example of use.

```
REPEAT 100 [FD 0.1 DOT CORFIX POS]
```

Ian suggests that the above values of YCOR (119, -120) in CORFIX are valid only for a SCRUNCH of 0.8, and Apple Logo II users would have to experiment to find the minimum and maximum YCOR values for SCRUNCHes other than 0.8.

Logo Lessons: A Guided Curriculum is available from John Glantz, PO Box 39, Liberty, PA 16930. Write for more information.

Sterling Swift announces the release of *Little Logo*, a package designed for young children with no computer experience, and *Computing in Logoland*, a self paced Logo tutorial course. For more information, write to Sterling Swift, 7901 South IH-35, Austin, TX 78744.

Are you using TI Logo and need to get your TI 99/4 or 99/4A serviced? Call (800) TI-CARES for information.

Tipps for Teachers

by Steve Tipps

Overland Mapmaking Adventures

Spatial understanding has been explored in a number of ways in the past four columns. Drawing pictures with dots, making maps of the classroom or neighborhood, and creating a geoboard with Logo are different ways of emphasizing spatial relationships. Understanding of space and measurement are important prerequisites to understanding and using maps of cities, states, and nations.

Many geography curricula begin with the description and exploration of local neighborhoods, and progressively move to places which are more distant. The ideas of towns, cities, counties, states, and nations as more complex, and comprehensive units of government may be difficult for children. Maps, as representations of these units, may be very remote to them.

Coloring or drawing maps and map puzzles are usual activities for elementary children. The use of Logo is another way to explore ideas about maps. By creating maps, children become familiar with geographical ideas.

Simple States

Although most states, provinces, counties and parishes do not come in regular square and rectangle shapes, simple shapes provide a good place to start drawing maps. Students can create their own imaginary states with the shapes they want. For example, the state of Centralia has the simple shape of a large rectangle, and is located in the middle of the screen.

```
TO CENTRALIA
PU HOME LT 90 FD 100 RT 90 BK 70 PD
REPEAT 2 [ FD 140 RT 90 FD 200 RT 90 ]
END
```

Students can name and draw their states. Octagonia and Triangula make good states. Jamesylvania and New Stacy are probably more exciting names.

Moving Around

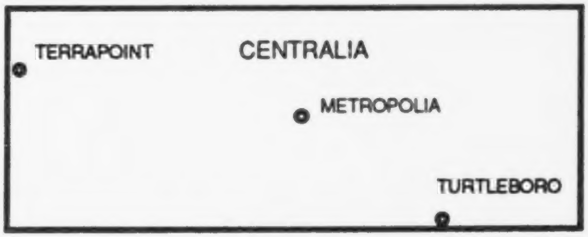
After the outline map is drawn, cities and towns can be added at selected locations. In last month's column, positions on the logogeoboard were designated with procedure names such as A2 or C5. When one of these procedures was called, the turtle moved directly to the corresponding spot on the geoboard. The same strategy can be used with place names.

The capital city of Centralia is called Metropolia and is located in the exact center of the state at position 0 0 (also known as HOME). A procedure for going home to Metropolia would move the turtle to that position.

```
LCSI versions           MIT versions

TO METROPOLIA           TO METROPOLIA
SETPOS LIST 0 0         SETXY 0 0
END                     END
```

More cities can be added to the map with new procedures. Perhaps small circles could be drawn to mark the locations of the cities.

```
TO TURTLEBORO           TO TURTLEBORO
SETPOS LIST 50 -70      SETXY 50 (-70)
END                     END

TO TERRAPOINT           TO TERRAPOINT
SETPOS LIST -100 20     SETXY -100 20
END                     END
```



As many places on the map can be added as students want to add. With each new city, they learn more about the coordinate system. Labeling the map with place names is typical on maps, and children will want to do this. Commodore Logo and IBM Logo have commands built-in to do this, but you may need the NLX alphabet for other versions.

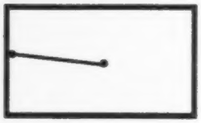
Tipps continued

Lines between the cities become connecting roads. Which of the cities are big ones and which are small? Which roads are likely to be the most traveled and the least? Transportation is an important item in most elementary social studies curricula. By the fourth grade, students may be ready to talk about the importance of roads in the delivery of goods. Study of the Interstate Highway System is a natural extension of these road building experiences.

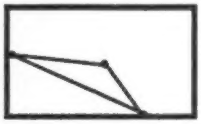
Highway Networks

Another extension of the roads drawn from city to city involves the study of networks. By starting out with only two cities, students can see that only one road is needed between them. With three cities, three roads are needed. How many roads are needed if you have four cities and want a road from each to each?

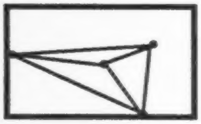
Number of Cities	Number of Roads Needed
2	1
3	3
4	______
5	______
6	______



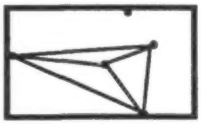
2 cities, 1 road



3 cities, 3 roads



4 cities, ___ roads



5 cities, ___ roads

What about five, six, or ten cities? Although the theory of networks is not included until high school or college, the experience of making all the possible combinations of road connections is quite within the grasp of elementary students.

Geographic Features

A map is not really interesting unless some geographic features are added. Mountains, rivers, lakes, and forests are easy to build in modular form and added to the map. Students have control over all these elements and can place them anywhere they want. A blue line makes a good river, for example. The line can be placed specifically or generated randomly. Since most rivers are not straight, the random river generator is a lot of fun to design and run.

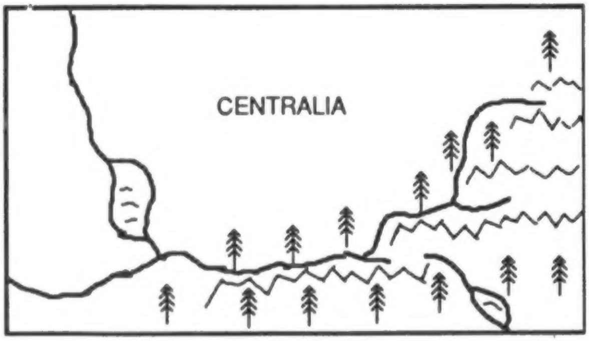
```
TO RIVER
REPEAT 10 [ FD (10 + RANDOM 10)  RT (30 - RANDOM 60) ]
END
```

RANDOM in this procedure is used with offsets. Students want to know why you have added and subtracted from RANDOM. They usually start with a river procedure which includes only RANDOM numbers as inputs to FORWARD and BACK.

```
TO RANDOM.RIVER
REPEAT 10 [ FORWARD RANDOM 20 RT RANDOM 30 ]
END
```

Let students begin with a procedure like RANDOM.RIVER and find out what difficulties it presents in make a good looking river.

Mountains can also be drawn with a procedure using random. Glen Bull has an activity in *Nudges: Apple Logo Projects* (Holt, Rinehart and Winston, 1985) which creates the Blue Ridge Mountains in a random fashion similar to RIVER. Students might also try a random curve procedure to make rivers and mountains. Mountains can also be small triangles arranged to show a mountain range. A small tree can be drawn. Several of these trees would represent a forest.



Rivers and Roads

After the students have completed all their geographic features, they might try drawing the roads again. If they have added rivers and mountains and forests and lakes, they may find that the most direct route between places on the map poses environmental problems. What if the road goes right through the forest? What if the shortest route must go over the mountains or across the middle of a lake? The dilemmas are just like the adult problems involved in road construction.

Tipps continued

See the USA

Whether you want to try drawing real states with Logo is conditional on the time and patience of the students. (See Rebecca Poplin's "Teacher Feature" in this NLX issue for details of a class Logo project to draw the various states.) Although Wyoming should be an easy state, New York and Florida are quite a challenge. Students tend to get bogged down in exact drawings. These may take dozens or hundreds of little turtle moves. Instead, have students compare many different maps of the US or Canada to see how artists have "simplified" the shapes of the states. They can also compare the detail of a large map to the detail of a small map. From these comparisons, they should realize that it is the overall shape of the state that is important, rather than every little bend in a river or coast line.

Students may also notice that many states west of the Mississippi are more regular in shape. This observation leads to an interesting research problem involving the boundaries of the eastern versus the western states. How did they get their shapes? Why?

Some students draw their states in a "free turtle" style. By experimentally turning the turtle this way or that, the state is completed. Another way is to use a transparency overlay as a grid over a map. Put the transparency over the screen and duplicate the distances. Some states may also need to be drawn modularly. Hawaii, for instance, could have a different procedure for each major island. California might have modules such as NORTH.COAST, OREGON.BORDER, NEVADA.BORDER, ARIZONA.BORDER, MEXICO.BORDER, SOUTH.COAST.

Using the legend of the real map is another aspect of map reading which should be stimulated. A little table stands for a road side park and an airplane is an airport icon. Students may want to add more features to their maps.

Maps are Everywhere

Logo is another way of alerting children to the meaning and usefulness of maps. The invented states, mountains, cities, and roads have real counterparts that are represénted on actual maps. Maps can also be found with varying amounts of complexity.

Maps are a primary method of communicating spatial information. Children need to be familiar with maps of their nearand far environments. By creating and playing with personal maps, their use and enjoyment of actual maps should increase.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Judith Harris

Would you like to probe the mysteries of Halley's Comet, study the intricacies of music theory, or create a hieroglyphic typewriter? These are just a few examples of interdisciplinary learning projects in which students and teachers who work with Judith Harris are learning to use the computer as a multi-faceted tool. Judith is the Originator / Coordinator / Specialist teacher in the "Computer as Tool Program" at the Solomon Schechter Day Schools of Philadelphia, Pennsylvania. She is an author, consultant, and computer education instructor in addition to working at least three days a week with students at all grade levels.

Out of This World

In a just-completed solar system project, Judith and the science teachers really wanted the students to use up-to-date information about the solar system, not just encyclopedia-based facts, and they also wanted students to understand that knowledge is not static. Using *Bank Street Filer*, the group created a data base with most appropriate information, including a field for Halley's Comet and information on the space shuttle missions. As new information is gained, students change or add to the data base.

One part of the project was a Solar System Walk for the whole school, starting, of course, in the computer lab. Questions were answered using FIND or SORT operations in the data base. Those who took the four-block-long walk, four billion times smaller than the real solar system, truly felt how close the inner planets are to each other compared to the distances between planets past Jupiter. Small groups of students who had conducted research on particular planets did presentations for the tourists and answered their questions. Later on, students in the two branches of the school will use a modem and *Newsroom* by Springboard to create and exchange a science and technology newsletter for the two branches of the school.

Logo a tempo

Judith, an active musician, has had some interesting experiences with Logo music. She used pattern blocks to help students who were experiencing difficulty with equivalent and mixed fractions. She played a song on a recorder as she showed them a sequence of blocks, explaining that the song and the sequence of blocks were related to each other. In fact, the different blocks represented the time values in the melody. She then had them mix up the pattern blocks and she played the song with the new timing. The students related their

Teacher Feature continued

new understanding to dealing with duration in Logo music. Judith found that Logo music makes it easier for kids to understand half steps and to transpose. Discovering how many sixteenths are equal to a quarter note is a perfect model for division of fractions.

Through Logo music, Judith also learned not to tell students "It can't be done." Two of her students really wanted to do harmony, and persisted, despite Judith's warning, in transcribing a piano duet, one working on each part, into Logo, and typing the resulting procedures into two computers side by side. The girls had thought that pressing the RETURN key of each computer at the same time would start the programs together and create harmony. To their dismay, garbage collections created pauses at surprising places, distorting the timing of the music. They discovered that using .GCOLL instead of R for rest minimizes the pauses and coordinated them somewhat.

Logoglyphics

Sixth grade students, studying ancient civilizations, had been doing various other projects with Egyptian hieroglyphs, so they created a hieroglyphic typewriter with Logo as a group endeavor. The kids divided the 55 hieroglyphs into easy, medium and hard, based on the difficulty of doing them in Logo. The hieroglyphs were divided among the groups of students, but they did share various types of arc procedures. Some used graph paper plans, but others did not. An important discovery was that all students used the strategy of scaling down, that is, designing the hieroglyphs large and then reducing them later to the desired size of 30 by 30 turtle steps. The students found that they could never use HOME and that they had to orient the turtle to the same starting point; they selected the bottom right corner of the square. They also decided that they didn't want the hieroglyph to fill the entire 30 by 30 step block; border space was necessary.

Finally it was time to put all the hieroglyphs together. The teachers had decided that a recursive structure would actually be best, but they felt the students weren't ready to deal with it. They had decided instead to separate the screen into lines of hieroglyphs. Imagine their surprise and delight when they were discussing the overall structure and the kids noticed the inherent patterns. One of the girls said, "Isn't there some way to get the procedure to make a copy of itself as it goes?" The need for recursion was invented! What a powerful idea!

In fifth grade, the curriculum has a heavy geography emphasis. Teachers approached the study in different ways but always seemed to have difficulty helping students understand latitude and longitude. A major Logo project creating maps of the states helped tremendously. Each state map had to display the locations of five cities using SETX and SETY as the equivalent of latitude and longitude. Students numbered their cities and made clues for each one, then challenged another group to identify the cities by clues. After using a National Council of Teachers of Mathematics activity which creates a different image of Morris the Cat by drawing a grid and distorting it, some groups extended this process to the maps. The students are now deeply involved in the process of cartography.

Another exciting development has been that classroom teachers observing Judith's work with students have begun to examine their own teaching and questioning processes. They are gaining a deeper understanding of teaching for thinking.

One group of fifth grade students is working with formal Euclidean geometry through Logo. Other students are studying truth tables through Logo. These are just a few of many interdisciplinary projects which are taking place at the Solomon Schechter Schools. If you are interested in further information about these or any of Judith Harris's other activities, she would be happy for you to contact her.

Judith Harris
104 Woodside Road, Apt. B-306
Haverford, PA 19041

If you know of teachers who are involved in exciting Logo projects that should be shared with NLX readers, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, TX 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Logo Publications

Logo in the Schools, a special double issue of the journal *Computers in the Schools*, contains 30 articles by Logo workers of many different orientations. This highly recommended volume of over 300 pages is available in softcover for $7.95, with a minimum order of 5 or more copies. If you are involved with Logo teacher training, this is an excellent resource to consider. For more information, contact Haworth Press, Inc., 28 East 22nd Street, New York, NY 10010, or call (212) 228- 2800.

Holt, Rinehart and Winston's series of books for IBM Logo includes *Nudges: IBM Logo Projects*, by Steve Tipps et al., *Exploring IBM Logo: A Guide for Adults*, by Sharon Burrowes et al., and *Beyond Mindstorms: Teaching with IBM Logo*, by Joyce Tobias, et al. An Apple version of *Nudges*, by Glen Bull et al., has been recently published. Also available is *Logo Physics*, by James Hurley. For more information, write to Holt, Rinehart and Winston, 383 Madison Avenue, New York, NY 10017.

NLXionary

A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"Interview: Seymour Papert," by Ron Schultz, *Omni*, October 1985.

>Feeling a sense of purpose. Seeing the big picture. Understanding the philosophy that guides one's daily efforts.

>What am I talking about? My reactions to Seymour Papert's comments in this lengthy *Omni* interview for their special 7th anniversary issue. If you find yourself languishing "spiritually" at times in your daily efforts with Logo or education in general, you might just get a needed boost from reading this article. It's heady stuff, though not complex. Read it slowly when you're well rested.

>Following a tidy biographical sketch of Papert, with a brief history of Logo, the interview begins with a discussion of Piaget, math, and microworlds. Papert argues that children who like to plan and decide exactly what should happen and then make it happen are more comfortable with typical school math because it emphasizes detail and rightness while minimizing play. Kids who like a more negotiable, consensual, interactive way of thinking have a harder time with math in typical schools. A computer with appropriate software can enable a student with this style to approach math in a "softer" context, allowing for self-determined manipulation without a right and a wrong.

>I love the metaphor for this line of thinking. "This is like a painter approaching a canvas with a general plan but without an absolutely worked-out top-down detailed anticipation of what it is going to be. You do something, look at it, stand back, do something else. It grows into your final product. School math doesn't let you do that very well. People who like to learn and master the world in this negotiational style can be very uncomfortable with the way that school makes you learn. For them, it is a harsh environment. Not for everybody."

>Reading frequently experiences a similar fate. Theories of decoding generate arguments about whether it is best to learn to read syllable by syllable, or with words as a whole. Papert says we seldom concern ourselves about children's *feelings* towards reading and writing. "The really important thing about learning to read and write is to love it!" (How many times do you hear of a school assessing their reading program by measuring the amount of reading for pleasure children do?

>In the same vein, he reminds us how much children are typically involved when doing art, and how this contrasts with their alienated approach when learning number facts or other basic skills. He believes that all learning can have the kind of involvement you see in art class. An aesthetic intent pervades Logo's turtle graphics environment, even though "what's happening on the screen is all sorts of geometry, number working, and other things you'd call math ... Aesthetic intent gives the integration its driving force and is, I think, a deep root of intellectual drive ... Schools also wrongly separate the aesthetic from the conceptual and so destroy this driving force of internal motivation." For those of you following the right / left brain learning movement, some of this should sound familiar.

>Computers can help fill in the gaps in what Papert calls *fractured knowledge*. He maintains that we cannot convey an idea to someone in the exact form that works for all of us. All we can give are pieces, and provide an environment where the person can build something with them. Computers can be used to provide the kind of environment that fosters experimenting and piecing things together. For example, he says if you do a lot of sailing, you've got a rich environment for understanding angles. Many abstract things we expect kids to learn are difficult because they have no frame of reference. Computers are flexible enough to provide children with many of those missing references. And Logo's many faces are a prime example of this flexibility.

>I've touched on only about half of the topics Papert covers in this interview. Some of the others may stimulate your interest more than they did mine. If you need to come alive a bit, and believe that getting in touch with your own sense of mission and vision can help, read the interview.

• • • • •

"Logo: A Language for Thinking about Thinking" by Robert Scarola, Learning Domain Editor, *Whole Earth Software Catalog for 1986*.

>The book is a gold mine, and Scarola's Logo contributions make it so. It contains a potpourri of Logo theory, books, and Jim McCauley's comments on all current implementations.

If you would like to respond to these articles and commentary, or know of other articles which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Listful Thinking

by Glen Bull and Paula Cochran

Anatomy of a Procedure

A new year is a time to consider new beginnings. This year, we're going to begin by showing you what the inside of a procedure is really like. Usually this column is focused on Logo applications in English and other subjects. However, many of the applications we develop are based on Logo's list processing capabilities. This month, we thought we would explore the structure of Logo, and show you how it is put together.

A procedure to make a SQUARE looks like this in the Logo editor. (Type in the procedure at this time.)

```
TO SQUARE :SIZE
REPEAT 4 [ FD :SIZE RT 90 ]
END
```

What does this procedure look like to Logo? To find out, we'll use the TEXT command.

```
PRINT TEXT "SQUARE
[ SIZE ] [ REPEAT 4 [ FD :SIZE RT 90 ] ]
```

A procedure is really a series of lists! (Of course! Isn't everything in Logo a list?) The first list contains the names of variable inputs, such as [SIZE]. Each line of the procedure is another list.

The Text of a Procedure

Let's verify this with another procedure. Here's a Logo procedure to make a house. It looks like this in the Logo Editor. (Type it in now, since you'll need it in a moment.)

```
TO HOUSE
SQUARE 50
FD 50 RT 30
TRIANGLE 50
LT 30 BK 50
END
```

You'll also need a triangle procedure. This probably is familiar to most of you.

```
TO TRIANGLE :SIZE
REPEAT 3 [ FD :SIZE RT 120 ]
END
```

Now use the TEXT command to discover the true nature of the HOUSE procedure.

```
PRINT TEXT "HOUSE
[] [ SQUARE 50 ] [ FD 50 RT 30 ] [ TRIANGLE 50 ] [ LT 30 BK 50 ]
```

Once again we see that each line in the procedure is actually a list. The first line in the procedure is [SQUARE 50]. The next line in the procedure is [FD 50 RT 30], and so on. The set of empty brackets [ ] at the beginning of the procedure is for variable names (if HOUSE had any variable inputs).

The Basic List Machine

Once we know that a procedure is really a series of lists, a whole realm of possibilities appear. Before we go on, we need to show you how to make a basic list processing machine.

In cooking, a basic mix or sauce is frequently used as the base for a wide range of other recipes. For example, many recipes start with a "basic white sauce". White sauce is the basic ingredient for other sauces ranging from Florentine sauce (for Eggs Florentine) to Mornay sauce.

In the same way, there is a basic machine which is used to process lists. It appears in a variety of guises, but generally looks like this:

```
To Process Lists

    (1) If the list is empty, stop.
    (2) Do something to the first item in the list.
    (3) Process the remaining items in the list.
```

Let's look at a Logo procedure which performs the steps indicated above. One of the simplest list processing machines inspects a series of lists and prints the contents one at a time.

```
TO INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

If you have Terrapin Logo, use this version of INSPECT.

```
TO INSPECT :LIST
IF :LIST = [ ] STOP
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

As you can see, the INSPECT procedure follows the three steps of a basic list processing machine. In the first line, it checks to see if the list is empty. In the second line, it prints the first item in the list. In the third line, it goes on to inspect everything else in the list.

Processing Procedures

Let's put the INSPECT procedure to work.

```
INSPECT [[ APPLE PIE ] [ ICE CREAM ] [ A B C ]]
APPLE PIE
ICE CREAM
A B C
```

As you can see, it prints out each item in the list on a separate line. Now apply INSPECT to a procedure. First, look at the TEXT of HOUSE to remind ourselves that it's just a series of lists.

```
PRINT TEXT "HOUSE
[] [ SQUARE 50 ] [ FD 50 RT 30 ] [ TRIANGLE 50 ] [ LT
30 BK 50 ]
```

Then inspect the lists in the procedure.

```
INSPECT TEXT "HOUSE

SQUARE 50
FD 50 RT 30
TRIANGLE 50
LT 30 BK 50
```

(Note that the empty brackets [ ] at the beginning of the TEXT "HOUSE list is INSPECTed and printed as a blank line.)

An Adventure

A colleague recently learned Logo, and spent six weeks constructing an adventure story. The story contained considerable text. One of the procedures looked like this:

```
TO INTRODUCTION
PRINT [THIS IS THE STORY]
PRINT [OF THE ODYSSEY.]
PRINT [ALONG THE WAY YOU]
PRINT [WILL HAVE THE CHANCE]
PRINT [TO MAKE SOME RIGHT]
PRINT [CHOICES AND SOME]
PRINT [WRONG CHOICES.]
END
```

If you know that a procedure is only a series of lists, another way to print text is possible. First, use the Logo editor to write the story, just as though it were a word processor. Press RETURN at the end of each line.

```
TO INTRO.TXT
THIS IS THE STORY
OF THE ODYSSEY.
ALONG THE WAY YOU
WILL HAVE THE CHANCE
TO MAKE SOME RIGHT
CHOICES AND SOME
WRONG CHOICES.
END
```

Then use INSPECT to print the text in the procedure.

```
INSPECT TEXT "INTRO.TXT

THIS IS THE STORY
OF THE ODDESSY.
ALONG THE WAY YOU
WILL HAVE THE CHANCE
TO MAKE SOME RIGHT
CHOICES AND SOME
WRONG CHOICES.
```

Knowing that a procedure is a series of lists can save hundreds of PRINT statements. It is only necessary to type PRINT once, in the middle of the list processing machine INSPECT. (For more information and details about the use of INSPECT, see the Listful Thinking column, "Power Tools," in the February 1985 NLX.)

Florentine Sauce

The INSPECT procedure is like the basic white sauce recipe. Let's modify it to make the Logo equivalent of Florentine sauce: a list machine which will allow a child to watch each successive step of a procedure "magically" happen each time the space bar is pressed.

First, change INSPECT so that a new line of text appears each time the space bar is pressed. This is accomplished by the SPACE.BAR procedure below.

```
TO SLOW.INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
SPACE.BAR
SLOW.INSPECT BUTFIRST :LIST
END

TO SPACE.BAR
IF RC = CHAR 32 [ STOP ]
SPACE.BAR
END
```

Listful Thinking continued

If you are using Terrapin Logo, use the following versions of SLOW.INSPECT and SPACE.BAR.

```
TO SLOW.INSPECT :LIST
IF :LIST = [ ] STOP
PRINT FIRST :LIST
SPACE.BAR
SLOW.INSPECT BUTFIRST :LIST
END

TO SPACE.BAR
IF RC = CHAR 32 STOP
END
```

The SLOW.INSPECT procedure works just the same as the INSPECT procedure, except that it necessary to press the space bar between each line. Try the SLOW.INSPECT procedure with the text of the HOUSE procedure.

```
SLOW.INSPECT TEXT "HOUSE
```

Did you have to press the space bar before each line of the procedure appeared on the screen? (You probably had to press the space bar before anything happened, because of the empty brackets [ ] at the beginning of TEXT "HOUSE.)

If the SLOW.INSPECT procedure worked, we're ready to go on to Mornay sauce. Mornay sauce is just a slight twist on white sauce. Our next procedure, which *executes* a procedure one line at a time, is just a slight variation on SLOW.INSPECT. Simply replace the word PRINT with RUN.

```
TO EXECUTE :LIST
IF EMPTYP :LIST [STOP]
RUN FIRST :LIST
SPACE.BAR
EXECUTE BUTFIRST :LIST
END
```

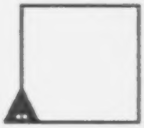
If you are using Terrapin Logo, use the following version of EXECUTE.

```
TO EXECUTE :LIST
IF :LIST = [ ] STOP
RUN FIRST :LIST
SPACE.BAR
EXECUTE BUTFIRST :LIST
END
```

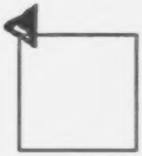
Try out EXECUTE with the HOUSE procedure.

```
EXECUTE TEXT "HOUSE
```

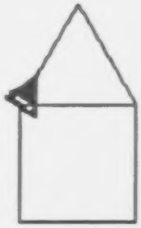
After typing in the line above, press the space bar. The turtle should draw a square.



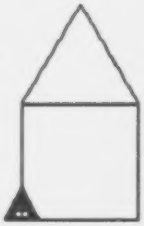
After the next press of the space bar, the turtle will shoot up to the top of the square.



After the next press,
the turtle draws the roof.



When the space bar is pressed
again, the turtle goes back down
to the bottom of the house.



The Captain Kangaroo television show used to have a magic slate, in which a drawing slowly appeared on the television screen as a story was read. The EXECUTE procedure can be used to create a similar magic slate, in which more parts of a drawing appear after each press of the space bar. (Many versions of Logo have a built-in command to allow you to execute a procedure one line at a time. Look for a name such as TRACE or STEP in the index of your Logo manual.)

Random Procedures

In Logo workshops, teachers frequently ask us for a way to randomly pick a procedure, and then run the procedure. If you are aware that a procedure is just a series of lists, it's easy to do this.

First we need a procedure to pick an item randomly from a list. Past readers of this column will be familiar with this procedure. (If you are using Version 1 of Terrapin Logo, you will need to write procedures for ITEM and COUNT. These procedures are found at the end of the Listful Thinking columns in the September, October, and November 1985 NLX.)

Listful Thinking continued

```
TO PICK :LIST
OP ITEM (1 + RANDOM COUNT :LIST) :LIST
END
```

The PICK procedure allows us to randomly pick an item from a list.

```
PRINT PICK [[APPLE PIE] [ICE CREAM] [A B C]]
ICE CREAM
```

Since we know that a procedure is just a series of lists, we can use PICK to pick a line at random from a procedure.

```
PRINT PICK TEXT "HOUSE
FD 50 RT 90
```

(Since PICK chooses randomly, it may pick a different line of HOUSE for you than it did for us. It may even pick the empty brackets.)

Create a procedure that contains a different shape on each line. You will have to write procedures for the PENTAGON, HEXAGON, and OCTAGON shapes.

```
TO SHAPES
TRIANGLE 50
SQUARE 50
PENTAGON 50
HEXAGON 50
OCTAGON 50
END
```

Now randomly pick different shapes from the SHAPES procedure.

```
PRINT PICK TEXT "SHAPES
PENTAGON 50
PRINT PICK TEXT "SHAPES
OCTAGON 50
```

To execute the picked procedure instead of printing it, simply substitute RUN for PRINT. This is the same trick that we used to change INSPECT into EXECUTE.

```
RUN PICK TEXT "SHAPES
```

What happened? Logo should have randomly picked and drawn a shape. Once again, one small change (substituting RUN for PRINT) allows us to obtain a different effect.

A Gourmet Logo Cookbook

What else can you do with the text of a procedure? We discussed ways to use a list of stars to create a star map in the November 1985 Listful Thinking column. The list of stars could be placed in a procedure. (Astute readers will note that the SPRINKLE procedure to sprinkle stars in that column is just a slight variation on the INSPECT recipe. The only change is substitution of DOT in place of PRINT.)

Some versions of Logo provide a mechanism for playing musical notes. Substitution of the command for playing a note in place of PRINT can change INSPECT into a procedure for playing notes. This revised INSPECT procedure might be called PLAY. Then a procedure of musical notes can be constructed.

```
TO TWINKLE.SONG
C C G G A A G
F F E E D D C
etc.
END
```

The procedure might work like this:

```
PLAY TEXT "TWINKLE.SONG
```

A Context for a Contest

Each month Robs Muir presents an NLXual (Intellectual ... get it?) Challenge. We thought that we were missing out on all the fun, so this month we are having a Logo contest of our own.

The National Logo Exchange will award a Koala Pad for the best procedure which makes use of the fact that a procedure is just a series of lists. Send your procedure listings to:

Glen Bull and Paula Cochran
University of Virginia Curry School of Education
109 New Cabell Hall
Charlottesville, VA 22903

The deadline for submissions is February 28th, 1986. The winner will be announced in the April NLX. We have decided to call this contest the N-L-TEXT-ual Challenge. Eat your heart out, Robs!

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is a doctoral student in the Curry School of Education at the University of Virginia. She is interested in Logo applications in language arts and speech-language pathology.

TurtleTips

by Jane Toth
and Donna Lanyi

"Isn't there an easier way to do this?"

Have your students posed this question to you after repeatedly (no pun intended) typing in the same commands? If so, this is the signal that the time is ripe to introduce the REPEAT command. Students are always excited about being let in on a "shortcut," especially if they have had to "do it the long way" for a while.

The Turtle Teacher

Students love using the REPEAT command because it cuts down on typing. We have found that they sometimes have difficulty deciding when to use this command. In order to help the children with this idea, we like to give them opportunities for discovering patterns.

An excellent precomputer activity suggested to us by Pat Arnold, a first grade teacher here in Wooster, was the use of string and different colored and shaped beads. Pat has her class sit around her in a half circle. Each student has a set of beads and string. Pat creates a pattern using colors first. She then has her students copy the pattern.

Of course, Pat encourages the children to verbalize the patterns; this reinforces the color concepts. After they've mastered color patterns, she leads them into shape patterns. She says that thay have difficulties at first in determining shape patterns but that someone eventually gets the ball rolling. After many experiences with this activity, she starts to mix the patterns with color and shape. What a challenge to those first graders!

We like our fourth graders to identify patterns from their life experiences. In one of our activities, we arrange students in patterns according to hair color, shirt color, types of clothing, or gender. We have them analyze the pattern and write it down. We also look at patterns in material. Students love to identify color striped patterns in shirts and jeans. After identifying many patterns, the students are encouraged to create their own color and shape designs using graph paper.

After studying patterns for a while, it is time to extend the activity to the computer. On the chalkboard, we have the students write the commands to create a square. Next, we ask them to look for the commands that have been repeated. How many times did we repeat those commands?

Then, we show them the form for using the REPEAT command:

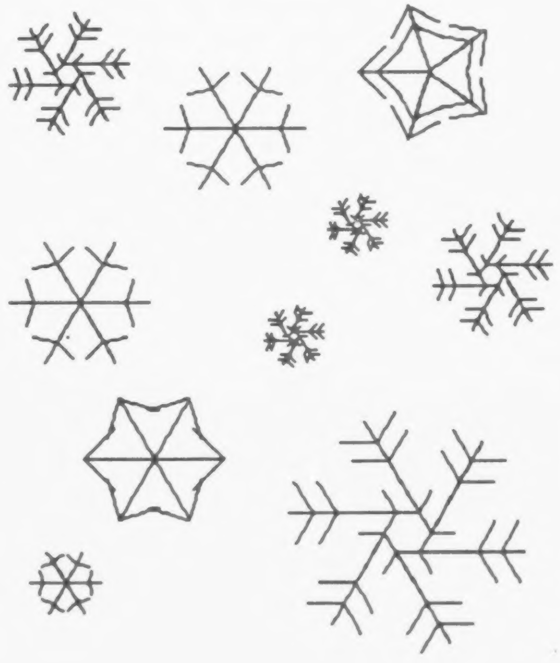
```
REPEAT (# of times) [ commands to be repeated ]
```

We brainstorm for other ideas for which we might be able to use REPEAT. Using a large screen monitor for demonstration, we have the students type in various REPEAT commands to explore what happens. Next month, we'll talk about having our students use the REPEAT command to make circles and arcs.

Procedure of the Month

Do you like to have objects hanging from your ceiling as a decorations? Why not go with the traditional snowflake, only this time from the computer? No two snowflakes are alike. Ask your students to create their own. Print them out, have the students color them in, and hang them from the ceiling. For those of you in the warmer climates, what better way to bring a bit of the chilly north to your classroom?

If you are interested in a copy of our snowflake procedures, please send us a self-addressed stamped envelope.



We know that many of you have fantastic ideas for Logo procedures and projects. Let us help you share them with others. Send your procedures or ideas for projects to Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691, or by electronic mail to us on CompuServe at 72477,2623.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher to Teacher

A Review of New Logo Materials

by Regina M. Bowden

Logoworlds, by Rachelle S. Heller, C. Dianne Martin, and June L. Wright, 1985, published by Computer Science Press, 11 Taft Court, Rockville, MD 20850.

Target audience: elementary and middle school teachers who are computer novices

Description: *Logoworlds* is a 301-page comprehensive collection of concepts, procedures, teaching tips, and "quidnunc" (Latin for "what now") suggestions to round out guided discovery modules: For complete computer novices, there is a table of Logoworlds, or subsets of Logo, identified with new primitives and corresponding activities. For example, the first Logoworld listed is "Body Logo." The new primitives are Forward, Back, Right and Left. Activities involving Body Logo include human robot games, magnetic or stuffed turtles, and robot toys. These activities may be found in Chapters 2, 15, and 16. Some other Logoworlds are: Doodling, Graphics, Words, Languages, and Large Projects. With Logoworlds, the new user is able to conceptualize "modules" rather than viewing Logo as one large complex language. By building module upon module, the teacher (and students) become more and more immersed in the language in a natural way.

Strengths: The Logoworlds book is written well for its intended audience. The authors make use of the "guided discovery" method. They justify its use by comparing "discovery learning" (which requires unlimited computer availability to be most successful), and "guided learning" (which presumes each student will progress through the same learning tasks).

In contrast, the guided discovery method allows the teacher-learner to explore and become comfortable with a new concept, and then be challenged to "discover" through the "quidnunc" or suggested activities that incorporate the new concept into a larger problem. This seems like a fair compromise, considering the limitations of computers and time in the average classroom. The teacher could concentrate on one chapter and spend weeks exploring the "quidnunc" suggestions, which helps internalize the concept, rather than move quickly through the chapters.

Other applause goes to the well written chapter on preschool use of Logo, which includes off-computer activities, the chapter on the use of robots and Logo, and the chapter on large group projects. This latter chapter stresses the use of Logo in creating microworlds for exploring many different ideas in science, social studies, and language units. This reinforces the philosophy that Logo is a language to learn with, rather than a language to learn!

What new Logo materials and books do you find useful in teaching? We invite you to send information to the Logo Information For Teachers (LIFT) group, PO Box 5396, Plymouth, MI 48170, or send electronic mail on CompuServe to 76317,565.

Regina M. Bowden is a microcomputer trainer for the Chrysler Corporation and a co-founder of the LIFT group.

Logo Disserts:

Dissertations Dealing with Logo

by Barbara Elias

What kind of mathematical environment is created when children learn the Logo programming language? This question was investigated by Richard Noss and the findings may offer some useful insights for elementary teachers. Noss, using a naturalistic and process-oriented approach, studied the Logo learning of 118 children in grades 3, 4, and 5 for 18 months. Children from inner-city, suburban, and rural schools were involved. Data were obtained from observation / participation, printed files, interviews, children's writing comments, and discussions with teachers.

One purpose was to determine the kind of learning strategies children use in their programming activities and how these activities constitute a mathematical environment. A second purpose was to determine the effect of the programming experience on the understanding of geometric concepts and elementary algebraic ideas.

The study involved three phases. The introductory and programming phases took place during a school year. The third phase involved testing and case studies.

The introductory phase (3 months) provided the fundamentals of Logo programming. The approach permitted children to pose their own problems, and encouraged them to work together and to learn from each other. An Apple II computer and a floor turtle were used in each classroom. Teachers and the researcher provided assistance and introduced programming techniques as needed. During this phase, children adopted three main strategies to avoid failure:

Logo Disserts continued

a. *homing-in* or a "guess and test" strategy,
b. modification of goals rather than debugging (i.e., *degoaling*), and
c. self-imposed *restrictions* in terms of inputs, commands and / or problems to be solved.

Although little planning was evident, goal-directed programming moved toward exploratory programming as children acquired more programming techniques. Mathematical activities of this phase included ideas about length and angle, circles, and turtle-based arithmetic.

During the programming phase (6 months), children were able to acquire important programming concepts in procedures, iteration, subprocedures, editing and debugging, inputs, and recursion. Learning strategies observed during this phase included making sense of a new idea, exploring, and problem solving.

(These strategies are proposed as a learning model since characteristic forms of mathematical and programming behavior are associated with each.)

During the third phase, a paper-pencil test on geometry concepts was given to 84 of the children and to a matched control group. In addition, data on the learning of algebraic concepts were obtained from case studies of eight of the participant children.

The results suggest that learning to program in Logo can serve as an effective means of creating a mathematical learning environment and can provide some benefit in the development of the concept of angle in geometry and the concept of variable in algebra. Of particular interest is the suggestion that children need support in overcoming "learning plateaus" while programming. In this study, it was found that teacher help, with new ideas, was more successful when given in an exploratory mode instead of a goal-directed mode.

>>>Noss, Richard (1985) *Creating a Mathematical Environment through Programming: A Study of Young Children Learning Logo*. Ph. D. thesis, University of London, 1985. (ISBN 0-85473-238-1). Copies are available for £20.00 (air mail) or £15.00 (surface mail) from the University of London Institute of Education, Department of Mathematics, Statistics, and Computing, 20 Bedford Way, London, England WC1 HOAL.

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Q and A

by Jim McCauley

Q. In BASIC, I am familiar with measuring the amount of memory that is available in bytes. In Logo, memory is measured in nodes. What are nodes, and how do they work?

A. In order to do a good job of explaining about nodes in Logo, I must first tell you a bit (please, no pun) about BASIC. You are correct that BASIC measures memory space in terms of bytes. This is a handy way to measure in BASIC because the ordinary way of storing collections of information in BASIC is in a data structure called an array.

Many different kinds of information can be stored in arrays, and each usually demands that a fixed amount of memory be set aside to store each piece of information. For example, integers take two bytes each, floating-point numbers require four bytes, and strings often call for 256 bytes to be reserved. Regardless of the size of the string or number to be stored, the same amount of space must be set aside for each, because BASIC accesses information in arrays by making jumps in memory equal to the fixed number of bytes allotted to each element of the data structure. This can be a good thing, since information stored in an array can be accessed very quickly.

But it is not such a good idea for some tasks (especially string storage) because it can be wasteful of memory. Also, the maximum length of the array must usually be specified before data is put into it. This means that problems which need dynamic data structures (that can grow or shrink as needed) cannot be represented conveniently.

The Node Mode

In Logo, the usual way of storing collections of data is in linked lists. There are many ways to create linked lists, and some versions of Logo use several at the same time. I'll explain a fairly simple technique here.

Logo lists are made up of nodes. Each node is made up of four bytes organized in pairs. Each pair of bytes can represent any number from 0 to 65,536 (that's 2 to the 16th power). This means that each pair can be a *pointer* to a location in the computer's memory. The first pair may point to the FIRST object in a list, while the second pair always points to the BUTFIRST of the list. The first pair may also play other roles; it might store an integer, part of a floating-point number, or part of a word. The last pair can have special properties, too. It may point to a special value, called "nil," which can represent either an empty list or the end of a list.

If you stop and think about it for a moment, these nodes are very flexible. They can represent any information in Logo: numbers, words, or lists. They have another key advantage in that a node that points to the BUTFIRST of a list does not have to be right next to the

Q and A continued

object that it points to. It can be anywhere in memory. For some computational tasks, this is a great advantage. Lists can grow or shrink as the problem demands, and this is much more useful than using arrays of fixed size. Indeed, to use linked lists in most computer languages, you actually must write a simple list processor into your program!

The Price You Pay

This flexibility has its costs, though. Lists usually take up more space than arrays, and to get to a particular item in the list, you must go through all the ones "in front" of it. Also, Logo's usual way of computing things is to make many copies of the things it is computing with (particularly in recursion). This tends to fill up the memory quickly, and it's possible to run out of nodes to compute with before the program is finished. In many kinds of problems, the Logo interpreter can detect when it's running out of nodes and actually recycle the nodes that it doesn't need any more, turning them into "fresh" nodes through a process called "garbage collection" or recycling. You can usually tell when Logo is almost "out of space," because the program will stop altogether while the unneeded nodes are converted to fresh ones available for use.

Sometimes, though, even recycling won't help. This is especially true of recursive programs. If you find that you are running out of space, it's usually due to one of two things. You may have a buggy recursive procedure that doesn't have a good stop rule (an IF statement that tells it when to stop "recursing"). Or, you may be trying to write a program that is just too big and complicated for the number of nodes available in your version of Logo. In those cases, you can get some practice in developing debugging strategies, and in Logo file management.

Each month, I will answer selected questions in this column about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to: Jim McCauley, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a self-addressed stamped envelope. You may also send questions to me on Compu Serve at 70014,1136.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

Terrapin Materials Announced

Logo Works: Lessons in Logo, a set of teacher developed curriculum materials, is being offered by Terrapin, Inc. Student books ($29.95) and Teacher Manuals ($12.95) are available. For more information, write to Terrapin, Inc., 222 Third Street, Cambridge, MA 02142, or call (617) 492-8816.

NLXual Challenges

by Robs Muir

It's Only Words

National education policy oftimes appears to be based on large, ill-defined games that are extra-national in nature. John von Neumann called games such as these "zero-sum" games, because they always have exactly one winner and one loser.

The Space Race we lost; Sputnik got there first. In the meantime, we taught (or attempted to teach) New Math. The Race to the Moon we won, with a giant step for Neil Armstrong and company. During this game, we tried to revitalize science instruction in our schools. The newest game is already having an impact on school policy. It is called, "Which Nation's Schools Turn Out The Best Educated Students?" We might also call this the "Schools as Factories" game. The stakes? Leadership in technology for the next decade.

According to some policy analysts, we are currently losing to the Japanese. One proposed answer is to increase the amount of science and math instruction our students receive. I have no real problem with increasing science instruction in schools. I just wonder what we'll need to drop from an already overburdened daily schedule to make room for MORE SCIENCE.

What is science, anyway? I won't bore you with Webster's definition. But let me propose that science can be done in any part of a curriculum. Science isn't just a subject such as chemistry or physics; science is a *method of inquiry* that transcends curricular boundaries. It is just that we are unaccustomed to doing science in an English class.

What's in a Word?

Let's do some science on five-letter words.

What is a word? Last year, my sixth graders and I concluded that a word is any group of letters and symbols (with no intervening spaces) that has a meaning associated with it. This is true of both English words and Logo words. English, unlike Logo, is a natural language; there are some random words built of English letters that are decidely *not* English words. Let's build some Logo words (using LCSI Logo) that will build random five-letter English words.

```
TO RANDOMWORD
OUTPUT (WORD LETTER LETTER LETTER LETTER LETTER)
END

TO LETTER
OUTPUT SENTENCE PICK [A B C D E F G H I J K L M N O P Q
R S T U V W X Y Z]
END
```

NLXual Challenges continued

```
TO PICK :OBJ
OUTPUT (ITEM (1 + RANDOM COUNT :OBJ) :OBJ)
END
```

```
REPEAT 5 [PR RANDOMWORD]
```

```
TSQUT
CKXGE
QUARK
LSJGL
SMOTF
```

Some of these are clearly non-English! (English is, after all, fundamentally pronounceable!) But what about QUARK and TSQUT? In an informal poll taken last year, 70 percent of my 5th and 6th graders correctly identified QUARK as the *real* English word.

What rules govern the English-ness of five-letter groups? The placement of vowels may be a good starting point.

After this poll, I challenged my students (all 63 of them) to write Logo programs that would generate five-letter words that really were English. The rules were simple: generate 1000 distinctly different words with your program and calculate the percentage of those that could be found in an English dictionary (or be a recognized colloquialism). In this way, I could also introduce simple statistics; how often do 6th graders get exposed to statistics, especially in an English lesson?

Several students developed programs that utilized good techniques for building English-like words. They incorporated rules developed through experimentation that were not even introduced in the language text we were using. Without exception, these students learned more about the structure of language than they would have by simply reading the book. However, they never exceeded a success rate of 10% with their Logo programs.

This project was suggested by E. Paul Goldenberg during the course of work on his forthcoming book, *Exploring Language with Logo*. (Goldenberg, Feurzeig)

The best efforts of his high school students with this problem are in the 17% range. Can you or your students exceed this?

It may be helpful to refer to the "Listful Thinking" columns in the September and October 1985 NLX for some additional ideas for starters.

Send your solutions along with a stamped self-addressed envelope, to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, CA 91711

We will compile several interesting solutions and redistribute them to all contributors. Please do *not* send your 1000-word program runs! The procedure code and your percentage results will be sufficient. By the way, you may send solutions to any past NLXual Challenges at any time. There is no "deadline."

Robs Muir is a teacher in the Claremont (CA) Unified School District and Claremont Graduate School.

Logo 86 Announced

Logo 86, the third annual international Logo conference, is scheduled for July 9 - 11, 1986 at the Massachusetts Institute of Technology (MIT) in Cambridge, MA. The program will offer major speakers, presentations, poster sessions, and special interest groups.

The conference will be preceded by two days of Logo tutorials.

Papers for presentations and poster sessions are invited from both educators and especially students. Papers outlining the major features of presentations (not more than 3 pages) or poster sessions (not more than 1 page) should be submitted by **March 1, 1986** to:

Brian Harvey
Logo 86 Program Chairperson
Massachusetts Institute of Technology
E15-309, 20 Ames Street
Cambridge, MA 02139

The National Logo Exchange, copyright © 1986 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223.

Editor..Tom Lough

THE NATIONAL LOGO EXCHANGE
The Logo Newsletter for Teachers

Volume 4 Number 6 FORWARD 100! February 1986 Page 1

Logo Teaching: A Marriage of Theory and Practice

by Ceil Humphreys

"In addtion to learning Logo in this course, we will also examine several psychological theories of learning and cognition that have dominated program and curricular development in mathematics and science teaching."

A collective groan arose from the class. When I had signed up for Dr. Gary Tubb's graduate course, "Programming Languages for Education: Logo," all I wanted to do was learn more about Logo. The course was part of my masters program at the University of South Florida in Tampa. I hoped that I would be training other teachers in Logo someday.

What? More Learning Theory?

The unexpected inclusion of the learning theories was a real shock. I already had a degree in education, and several years of teaching experience. I had developed a few theories of my own; did I really need any more? Somewhat reluctantly, I accepted the theoretical aspects of this so-called "Logo" course.

We were scheduled to study Gagne, Bruner, Ausabel, Papert, and Piaget, in addition to learning IBM Logo. Each student was required to write a paper to show the use of Logo to teach a particular concept and to explain in detail the learning theory concepts selected and employed.

It wasn't until later that I discovered how important the learning theory part of the course would be in developing practical applications as a Logo teacher trainer.

One Year Later

The following year, I was given the opportunity to teach the Logo section of an introduction to computers class for undergraduate education majors. I had to cover many Logo concepts in a short amount of time, and still keep the students enthusiastic. I wasn't sure at all how I was going to do this.

After introducing the concept of procedures, for example, I needed an effective activity to reinforce the idea and give an opportunity for practice. The students needed to understand how to organize a block structure and to discover the advantages of dividing a large programming task into smaller subtasks. I wanted the activity to be fun, open-ended, relevant, and also give the students the opportunity to be creative. Finally, I wanted the students to produce hard copies of their projects for a classroom display.

Since it was October at the time, Halloween was on everyone's minds. I found an activity in *Teaching and Computers* magazine ("Carve Logo Pumpkins," October 1985, page 50) which I could modify to incorporate all of my learning objectives.

I gave the students procedures to draw a pumpkin body, a stem, and a triangle. Then I asked them to "carve" their own jack-o'-lanterns by using these Logo tools and writing other procedures for the eyes, nose, and mouth. I encouraged them to be creative with the color commands, and to write a superprocedure to organize the entire project.

Logo Teaching continued

Surprise, Surprise

It was at about this point I realized I had applied David Ausabel's concept of "advanced organizers" to my Logo teaching! An advanced organizer, I remembered from my Logo class work, is an organizing statement which links what the learner already knows with what is to be learned, and creates an organization or structure within which the new learning will be embedded. (See Ausabel, D. P. *The Psychology of Meaningful Verbal Learning.* New York: Grune & Stratton, 1963.)

The tool procedures I furnished the students served as a link to what they already knew, and the project assignment created the basis for the new learning specified by my additional instructional objectives.

As I reviewed what I had learned about Ausabel's theory in Dr. Tubb's Logo course, I became aware of many other possible applications at every level of the curriculum. It all seemed to "click," now that I had an applied Logo perspective. Suddenly, my Logo training design took on a whole new meaning!

By the way, when my students brought in their Logo jack-o'-lantern projects, I was amazed at the wide range of "personalities" of the pumpkins. Several students commented that the tool procedures and handout I had furnished were most helpful in giving them a clear starting point. It seemed that the advanced organizers certainly had done the job!

Yes, More Learning Theory

Although I learned about the various learning theories reluctantly in my Logo course, I now feel they have greatly improved my Logo teaching methods.

If you are a teacher trainer, I encourage you to consider adding more learning theory to your Logo training. This might be an excellent opportunity to team teach with a "more learned" colleague who has not yet investigated the possibilities of Logo.

I encourage all NLX readers to investigate Ausabel (and all the other important learning theorists) and explore the richness of their work from your Logo perspectives. This could add a new dimension to your Logo teaching, as it did to mine.

Ceil Humphreys is involved with Logo teacher training at Claywell Elementary School in Tampa, FL.

From The Editor

by Tom Lough

"In your presentation, you mentioned the NLX. What is the NLX? Do you recommend that I join? How can I get in touch with them?"

Have you ever been asked these questions? I hope so. How have you answered them?

The primary objective of the National Logo Exchange is to serve teachers who use Logo in their classrooms. We do this mainly through our newsletter. In the past three and one-half years, we have provided the Logo community with more than two hundred and fifty articles for this purpose. From your many letters of support and encouragement, it seems that we are meeting many of your needs.

Many citations now refer to the NLX. The newsletter appears on practically every Logo resource list. Yet, teachers sometimes do not know about the newsletter and have difficulty learning of our address.

In order for us to be effective and to serve the Logo teachers who need new ideas and up-to-date information, our address should be readily available. When teachers ask questions such as those above, informative answers should be furnished.

NLX readers are members of a very special family. You believe in the NLX and support its activities through your subscriptions. We appreciate that support very much. Would you be willing to help provide information about the NLX to others?

If you are giving a Logo presentation or putting on a Logo workshop, tell your attendees about the NLX. If you are teaching an educational computing course which includes Logo, consider informing your students about the NLX. If you are going to an educational computing conference, think about telling those you meet of the NLX.

In addition to telling others about the newsletter, you might want to augment your Logo handouts with NLX brochures. Make them available at your Logo presentations. Give them to your workshop attendees and to those in your Logo courses. Take them to conferences and place them on the "freebie" tables.

By helping to "spread the word" about the NLX, you would also be helping bring useful Logo teaching information to those who need it.

To order a free supply of NLX brochures, write to: *NLX Brochures, PO Box 5341, Charlottesville, VA 22905.* Tell us the number of brochures requested, the date needed, and a *brief* description of the activity at which they will be distributed. If you live in the USA, please provide a street address for UPS delivery.

It is always fun to tell others about a source of information which has been useful. We hope that you feel that way about the NLX, and will enjoy making NLX information available to colleagues in your own Logo community.

Then, when someone asks you about the NLX, you can give them a brochure ... and a smile!

Thanks for helping Logo and the NLX move ...
FD 100!

NLXtra

A Supplement to the National Logo Exchange Newsletter

Volume 4 Number 6 *FORWARD 100!* February 1986 Page 1

Imagewriter Overprinting Solved for Apple Logo II !

John Olive, director of the Atlanta-Emory Logo Project of Emory University, Atlanta, GA, has solved the problem of the overprinting with Apple Logo II and the Imagewriter printer! He has written a set of procedures which not only prints out Apple Logo II files but also allows the user to set the left margin and to specify the number of printed lines per page.

He reports, "I had been frustrated with the overprint problem for more than a year when I saw the solution offered for [Apple] Logo (October NLX). I tried all possible ways to send the suggested codes [in Apple Logo II] but was unsuccessful. It was only after looking at the PLOTSTUFF procedures (which drive the Apple Color Plotter) that I was able to figure out the correct sequence of commands for sending control codes to the [Imagewriter] printer via Apple Logo II. After solving the interface problem it was just a matter of deciphering the Imagewriter User's Manual to come up with the appropriate control codes."

Here are John's procedures for Apple Logo II.

```
TO SETPRINTER :LM :PL
OPEN 1
SETWRITE 1
ADDLF
SETLM :LM
SETPL :PL
SETWRITE [ ]
CLOSEALL
END

TO ADDLF
TYPE CHAR 27
TYPE CHAR 68
TYPE CHAR 32
PR CHAR 0
END

TO SETLM :N
IF :N < 100 [ MAKE "N WORD "0 :N ]
IF :N < 10 [ MAKE "N WORD "0 :N ]
TYPE CHAR 27
PR WORD "L :N
END

TO SETPL :N
( TYPE CHAR 29 CHAR 65 CHAR 64 )
REPEAT ( :N - 2 ) [ ( TYPE CHAR 64 CHAR 64 ) ]
( TYPE CHAR 67 CHAR 64 )
REPEAT ( 66 - :N ) [ ( TYPE CHAR 64 CHAR 64 ) ]
( TYPE CHAR 65 CHAR 64 )
PR CHAR 30
END
```

These procedures are written for the standard 80 column Imagerwiter printer with a Super Serial card in slot 1, and assume continuous feed eleven inch paper. They produce hardcopy output at 66 lines per page and 6 lines per inch.

Here is a procedure as an example of how to use SETPRINTER.

```
TO PRFILE :FILE
SETPRINTER 8 58
DRIBBLE 1
POFILE :FILE
NODRIBBLE
END
```

The file :FILE is printed out with a left margin of 8 characters and 58 printed lines per page. Once PRFILE is run, the printer keeps the same configuration for other text dumps using DRIBBLE 1 until SETPRINTER is used with different inputs or the printer is turned off.

Our congratulations and thanks to John for sharing these procedures. They will make life easier for Logo teachers everywhere. But John suggests an additional lesson from the experience. "Looking at procedures that solve similar problems is an important programming strategy which we should encourage in our classrooms." We add our hearty "Amen!!"

NLX readers wishing to contact John may write to Dr. John Olive, Project Director, Atlanta - Emory Logo Project, Emory University, Division of Educational Studies, 201 Fishburne Building, Atlanta, GA 30322.

Other Logo Printing Notices

NLX reader Eadie Adamson uses Applewriter IIe for typing in long programs for Apple Logo II. She then saves the file with .LOGO on the end of the file name, and calls it up from Apple Logo II. She credits Joe Delibero of Packer - Collegiate in Brooklyn, NY, with this idea. Eadie also uses Applewriter IIe to print out Apple Logo II files on the Imagewriter without overprinting. For more information, write Eadie Adamson, The Allen - Stevenson School, 132 East 78th Street, New York, NY 10021.

Have you been frustrated by the inability of Apple Logo to print pictures? Sharon Burrowes' article, "How Can I Print My Apple Logo Pictures?" in the December 1985 / January 1986 issue of *The Computing Teacher* (1787 Agate Street, Eugene, OR 97403) contains a number of useful techniques. Sharon also offers a disk containing programs to print Apple Logo pictures. For more information, write to Sharon Burrowes, 807 College Avenue, Wooster, OH 44691.

Nicole Michaud of Logo Computer Systems Inc., and Eric Nagler of Santa Clara, CA, sent in a procedure for printing Apple Logo II pictures with a Grappler+ card and a parallel printer. Nicole also sent instructions for printing Apple Logo II pictures with a Prowriter II printer. For a copy of either set of instructions, send a self addressed envelope to NLX (Grappler+ or Prowriter), PO Box 5341, Charlottesville, Va 22905-0341.

Terrapin Logo users who want the TEXTEDIT utility program to establish the top and bottom margins of printed pages will now be able to do so. NLX reader Lawrence Alexander has written a set of procedures for serial printers which can be added to TEXTEDIT to maintain top and bottom margins as well as to number each printed page. For a free listing of the Terrapin Logo procedures and documentation, send a stamped self-addressed envelope to Prof. Lawrence Alexander, College of Education, Michigan State University, East Lansing, MI 48824-1034.

NLX Logo ABC Book Planned

The NLX ABC Logo procedures to draw the letters of the alphabet in any size have been distributed free to over 3000 educators. Now it's time to compile the results! *The NLX Logo ABC Book* is in the planning stages, and will contain projects created with the NLX ABC procedures. If you and your students have used the ABC's in any way to create designs, drawings, or other applications, why not send in your work for inclusion in this book? Editor Michael Friendly reports, "I'm excited about the many different projects possible for the book ... drawings made with just one letter, drawings made with words, the list is endless!" Please send procedure listings for your ABC projects to Michael Friendly, 116 Wells Street, Toronto, Ontario M5R 1P3. Include a graphics printout if possible. If you have not yet received your copy of the NLX ABC's, send a stamped self-addressed envelope to NLX ABC's, PO Box 5341, Charlottesville, VA 22905.

Logo Notes

Logo versions are now available in a growing number of different national languages. For information on a particular language, write to Nicole Michaud, Logo Computer Systems, Inc., 9960 Cote de Lisse Road, Lachine, Quebec, H8T 1A1 Canada, or call (514) 631-7081.

Does anyone have any ideas for Logo applications in shorthand classes? If so, send your suggestions to Wanda Printis, 216 Lowden Hunt Drive, Hampton, VA 23666.

The Educational Computing Organization of Ontario (ECOO) announces its Logo conference, "Look to the Learner," to be held March 22, 1986, at York University in Toronto,Ontario. The program features such speakers as E. Paul Goldenberg, William Higginson, and Michael Friendly. This is a full day of Logo activity you will remember for a long time! For more information, write to Pat Davidson, Sloane Public School, 110 Sloane Avenue, Toronto, Ontario M4A 2B1.

The COMPSCI '86 conference will be held on May 12 - 13, 1986 in Clayton, Australia. Also, "Computers in Education: On the Crest of a Wave?" is the 4th Australian Computers in Education conference and is scheduled for August 24 -27, 1986. For more information on these conferences, write to the Computer Education Group of Victoria, PO Box 88, Balaclava, Victoria 3183, Australia.

Biller & Snyder announce the development of a powerful Logo environment to work with Macintosh Logo. For more information, write to Loren Abdulezer, Director, Management Services, Biller & Snyder, 75 Maiden Lane, New York, NY 10038.

Coral Software Corporation has announced the development of Object Logo, an advanced Logo version for the Macintosh. For more information, write to Coral Software Corp., PO Box 307, Cambridge, MA 02142, or call (617) 547-2662.

It's never too early for Christmas! Chuck Thompson has written a set of procedures which draws a beautiful set of tessellated Christmas trees in variable sizes. For a free listing, send a stamped self-addressed envelope to Chuck Thompson, Christmas Tree Tessellation, School of Education, University of Louisville, Louisville, KY 40292.

Tipps for Teachers

by Steve Tipps

Getting to Know You

Talking to other people and finding out things about them is an everyday human activity. When you meet someone for the first time, you usually ask about his or her name, home, occupation, and hobby. Logo can be used to simulate getting to know someone. In the process of converting human-to-human conversations to the computer, ideas about language, Logo, and human nature are explored.

Simple Questions

Have the children consider a situation in which they meet somebody for the first time. List the many things which they might ask. Writing the questions in Logo takes only a PRINT statement.

```
PRINT [ WHAT IS YOUR NAME? ]
PRINT [ WHERE DO YOU LIVE? ]
PRINT [ WHAT DO YOU LIKE TO EAT? ]
```

The questions are not much good, however, without answers to them. When you are asking a question of another person, you stop and wait for an answer. You listen with your ears and remember with your brain. Logo can also stop, wait, listen, and remember, using the command READLIST (or REQUEST). This command causes Logo to stop and wait for another person to type in a response to a question.

If you type the commands in by themselves, you may think that the computer has died. All you see is a blinking cursor. Logo is waiting for you to type something (anything!) on the keyboard. The cursor blinks until you type and / or press <return>. You need to know what to type. When you use READLIST (or REQUEST), be sure to make the program friendly by including instructions or directions to go along with them. Make a procedure which combines the friendly question wih the request for the answer.

```
TO ASK.NAME
PRINT [ WHAT IS YOUR NAME? ]
PRINT READLIST      (for MIT Logos, use PRINT REQUEST)
END
```

You have just the beginning of a conversation. The procedure asks the question and stops, waits, and listens for an answer. The difficulty is that the procedure only repeats your name.

The computer asks, "What is your name?"

If you type "Jamie," the computer also types "Jamie."

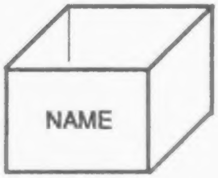
If you ask ten other questions, the computer just repeats your answer each time. The computer echoes your answer. This habit is irritating. In addition, the computer does not remember your name or address or favorite team or anything else you tell it. The program has no way to remember your answer. As the Scarecrow, it doesn't have a brain for remembering.

Making Boxes

The lack of a brain can be remedied by creating a place to hold on to the answer with a MAKE statement using READLIST (or REQUEST).

```
TO ASK.NAME
PRINT [ WHAT IS YOUR NAME? ]
MAKE "NAME READLIST    (MIT: MAKE "NAME REQUEST)
END
```

The effect of a MAKE statement is to create a place to put the answer. Many people like to think of this place as a box with a label on it. In this case, you have created a box with the label "NAME" on the outside. The box holds the name which was typed in with READLIST (or REQUEST).



Only one name can be in the box at any time. The last name typed in is the one which is in the box. So, every time you ASK.NAME, a new name goes in the "NAME box.

Careful Questioning

This one simple question form can be adapted to any question you want to ask. But you have to be careful that your question asks exactly what you want to know. Consider a procedure which wants to know which town you live in.

Tipps continued

```
TO ASK.TOWN
PRINT [ WHERE DO YOU LIVE? ]
MAKE "TOWN READLIST  (MIT: MAKE "TOWN REQUEST)
END
```

The answers you might get include "444 Main Street," "La Vista Apartments," "California," or "around the corner." To get precise information, ask precise questions. Make up a series of procedures each of which asks a single question and gets the answer. Each question involves a seperate procedure or module. Each procedure should do one task well.

Getting Nosy

All the procedures can be combined into a single program called NOSY which asks many questions of people.

```
TO NOSY
ASK.NAME
ASK.TOWN
ASK.STREET
ASK.FAVORITE.FOOD
ASK.FAVORITE.TEAM
ASK.BEST.VACATION
END
```

Students invent many, many questions depending on their interests. College students in an Introduction to Computer Science class worked up an elaborate NOSY program to screen prospective dates with such personal questions as kind of car, income, looks, and favorites of every kind. By working on separate procedures which did only one thing well, they were developing modular programming habits.

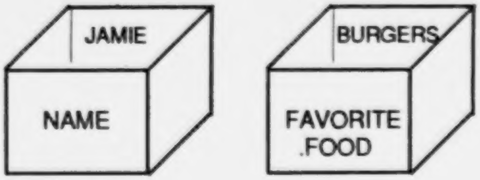
What's In the Box

Each of the ASK procedures makes a box and collects an answer in it. The procedure now has a brain, but needs a way to retrieve what it remembers. You may want to know what THING is in the "NAME box or what THING is in the "FAVORITE.FOOD box. In Logo, you PRINT the THING in the box.

```
?PRINT THING "NAME
JAMIE
?PRINT THING "FAVORITE.FOOD
BURGERS
```

A shorthand way of finding out what is in the boxes is the more commonly seen :NAME (with a colon, or "dots") instead of THING "NAME.

```
?PRINT :NAME
JAMIE
?PRINT :FAVORITE.FOOD
BURGERS
```



Both forms work equally well at retrieving the contents of the variable. However, our experience is that using THING as the referent to the contents of the variable box results in better understanding of what is going on. When students (or teachers) later use the short form, :NAME, they carry with them the mnemonic of "the thing in the NAME box." Without this experience, they are constantly confused about when to use "NAME and when to use :NAME.

"NAME is the label
:NAME is the thing in the box

In more advanced work with lists, a full understanding of THING is absolutely essential. By starting with THING, list processing is much easier.

Tell All

After asking all the questions and remembering all the answers, Logo is ready for a procedure to tell all about a person. You could just summarize the facts with simple PRINT procedures.

```
TO TELL.ALL
PRINT THING "NAME
PRINT THING "TOWN
PRINT THING "FAVORITE.FOOD
...
END
```

The difficulty is that the procedure prints out just the information and no description of what it is. A better way of doing this is with a SENTENCE which tells what is being printed.

```
TO TELL.ALL
PRINT SE [ YOUR NAME IS ] THING "NAME
PRINT SE [ YOUR TOWN IS ] THING "TOWN
PRINT SE [ YOUR FAVORITE FOOD IS ] THING "FAVORITE.FOOD
...
END
```

SE is the short form for the SENTENCE command. It is useful when you are going to print two things on the same line.

Tipps continued

Interviewer Extraordinaire

These questioning tools can be used in many ways in the classroom. One way is to have the student of the week answer the questions. Then turn on the printer to share important information with everyone in the class. Another use would be in language or social studies for collecting information about famous people. You could write an interview for George Washington, Laura Ingalls Wilder, or Ralph Nader. Fictional and fantasy characters such as Winnie the Pooh or E.T. might also be interviewed. Interviewing could be a creative writing stimulus. Older students might study interviewing techniques on TV. Remember the question, "If you could by any kind of tree (or animal or whatever), what would it be?"

At the end of the interview, students can also save all the information. One simple way is to save all the procedures and the current answers in a file.

```
?SAVE "GEORGE.WASHINGTON
```

Of course, all the procedures are repeated in every file which is handy but can use a lot of disk space. Another method is to erase the procedures using ERPS (or ERASE PROCEDURES) before saving the file. Then only the current answers are put into the file.

From a simple question to a whole interview requires only five Logo commands: PRINT, READLIST (or REQUEST), MAKE, THING, and SENTENCE. They are simple commands which lead to exciting possibilities in writing and reading. The real goal of getting to know you is working and playing with language just as the real goal of Logo is expressing ideas.

If you have additional ideas to share, please send them to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or on CompuServe at 76606,1623.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Resource Packet Available

The International Council for Computers in Education (ICCE) has for sale a Logo resource packet which contains copies of 2 Logo issues of *The Computing Teacher* magazine and reprints of nearly 20 additonal Logo articles. The packet makes an excellent supplemental reading resource for teacher training courses, and is available for $10.50 from ICCE, 1787 Agate Street, Eugene, OR 97403.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Peter Anderson

Logo teachers are explorers and discoverers, always searching for new ideas, new perspectives, and new understandings. Peter Anderson, a teacher from Sharon, VT, responded to his urge to investigate the uses of computers in education, and set out last year on an exciting odyssey many of us have only imagined.

On the Road

Peter was granted a year's sabbatical from his position as English teacher at the Marion W. Cross School in Norwich, VT. He used his time to visit schools and conferences all over the United States, England, Norway, and Sweden. Peter defined three major interests related to computers in education: (1) the effect of school policy on computer use, (2) teacher training, and (3) word processing: using technology to help students write.

Peter had been using computers for Logo as well as word processing in his classroom. He was fascinated with the creative stimulation of Logo problems and wanted to learn more. His first venture was to attend Dan and Molly Watt's Logo Institute in Keene, NH. (Ed. note: For information on this year's institute, write to Tony Stavely, Keene State College, Keene, NH 03431.) There, he worked with 40 teachers from all over the country. His experience raised many questions concerning what education is about and what the role of technology should be.

Metaphors Wanted

Meanwhile, he became fascinated by Logo's use of words and lists, but had difficulty at his level of programming experience. As Peter tried to devise a metaphor to help him understand how Logo processes lists, he decided to write a play for his fifth and sixth grade students called "Logo Processes a List."

Peter spent the fall visiting Harvard and MIT, attending seminars. He worked at Coleco for two weeks on the conceptual stage of software design. Another trip was to the University of New Hampshire to work with Donald Graves, an educational researcher who is studying the use of computers to promote writing in the early grades. Peter also went to the Minnesota Educational Computing Corporation (MECC) conference in Minnesota, and found others who were interested in his ideas for Book / net, an international publishing exchange.

Teacher Feature continued

Stooper Troops

In January, with his wife, Ina, and younger son, Angus, Peter packed his camper van and began to travel all over the United States. He soon learned that the van's height made it difficult to don jacket, tie, and briefcase for his interviews. Angus quickly dubbed him "Stooper Man," and the name stuck for the rest of the trip!

The first stop was in Washington, DC, to visit with Frank Withrow at the Department of Education. Peter was encouraged to find that they both shared many of the same ideas about computers in education. Visits to schools from Maryland to South Carolina convinced Peter that many school districts are still struggling with the implementation of computers.

He decided to visit the Florida Educational Computing Conference, because Florida had just created a statewide computer curriculum. What Peter found was that teachers had been given a set of computer competencies and were desperately visiting vendors to find software for specific computer skills. Attendance at the Texas Computer Education Association conference in Austin revealed a similar situation there. Peter spent time in Houston, learning about the school district's in-house software development team, where one of many products has been a computer based "English as a Second Language" program.

In Arizona, Peter visited with some educators who were very interested in evaluating software. He became even more convinced that there is a lifetime's worth of work with a good word processor, Logo, and a data base, and became more wary of passive or non-interactive software.

A sojourn in Bruce Fleury's classroom at the Magnolia School in El Cajon, CA, provided Peter with an example of a way to decentralize computer equipment and prevent what he calls the "shrine room mentality" apparent in many schools. On a roof-covered deck with access to several classrooms, a computer / printer setup was available for use by different groups of students. Peter also observed second graders being taught by fifth graders.

While in the San Francisco area, Peter did a workshop on publishing with computers, and shared what he had seen on his travels.

Up and Away

After a time back in Vermont, Peter left the US to pursue his interest in policy in countries with a centrally organized state educational system. In Sweden, he found that the government has commissioned educational software which reflects the Swedish culture and identity. Except for a single pilot program, Sweden has not started using computers in elementary schools. Teacher training is a major problem there, and it is a struggle to keep teachers in education. In addition, the education authorities want to study computers more extensively and proceed slowly.

In Sweden, Norway, and the United Kingdom, Peter made contacts for his Book/net project, and has since organized the participation of fifteen schools in ten different countries. Peter has published an article about his work, "Classroom Publishing with Micros," in the October 1985 issue of *The Computing Teacher.* Peter has been invited to return to Sweden in March to make a follow-up presentation.

Peter's exploration has led him to several conclusions about the state of computer education. One is that teacher training is a key problem everywhere, with no totally satisfactory solutions. He is encouraged that people no longer seem to have the notion that every child should learn BASIC, and that people are challenging the idea of the concrete curriculum. It became even more obvious to him that there are no easy solutions. We must see technology in education as an experiment, and be prepared to adapt as we learn more.

Send Your Cards and Letters

Peter would be interested in further contact with NLX readers concerning the Book/net, his travel experiences, workshops, or his Logo play.

Peter Anderson
Route 1, Box 444
Sharon, VT
(802) 649-1519
CompuServe 72437,2251

Sharing ideas is a big part of what the NLX does for teachers. If you have ideas you would like to share through this column, or you know of teachers who should be featured, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, TX 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Logo Challenge Offered

Apple Computer Clubs International have announced their Challenge Program, a merit award program similar to that of scouting merit badges. One of the eleven challenge areas is Programming in Logo. This is an excellent opportunity for students to design and carry out Logo activities which will help them learn even more. For more information, write to Challenge Program, Apple Computer Clubs International, PO Box 948, Lowell, MA 01853.

NLXionary
A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"The Rhetoric and Reality of Expanding Educational Choices," by Joe Nathan, *Phi Delta Kappan*, March 1985.

"The Synthesis of Research on Schools of Choice," by Mary Anne Raywid, *Educational Leadership*, April 1984.

>I digress.

>These articles have nothing to do with Logo, and yet they have everything to do with Logo. I bring them to your attention because I believe they address the philosophy of Logo. Although the subject seems far removed from everyday classroom activities, it directly affects classroom functioning in a multitude of unseen ways.

>The issue is permitting families to select from among different public schools. Why? I'll list some of my reasons. But first, think about Logo and its philosophy for student involvement. There are many components, but choice is a cornerstone. Without choice, exploration and discovery are hollow. We've all seen instances where the spirit of Logo was thoroughly undermined when implemented in a school district.

>If you look closely, lack of choice is usually at the center. It may have been that teachers had no choice about their participation in Logo's implementation. Possibly students were expected to learn Logo according to a prescribed scope and sequence - usually devoid of choice. Successful Logo project implementations almost by definition involve choice.

>But the structure of public education in America, land of freedom, uncharacteristically defines the parents' "choice" of where to send their children to public school as a function of where the parents happen to live or are able to live.

>This is the perfect recipe for creating uninvolved and unmotivated parents, just as in the schools for creating uninvolved and unmotivated students.

>The research indicates that when families are permitted to select from among different public schools:

- parental satisfaction with the educational system increases
- students' self-concept, behavior, attitudes toward learning and towards schools improve
- student attendance and achievement increases
- teacher morale goes up as teachers have opportunities to work in programs which fit their educational philosophy (Teachers also like the opportunity to use special expertise they've developed. One way to help make this happen is to provide faculty with opportunities to create programs and then permit families to select from among them.)
- community support for schools increases as people hear about diverse high quality offerings
- in the beginning, affluent families have more information about choices; however, as times goes on (and the rules stay the same), the difference in knowledge about choices decreases
- although "choice" can be misused by a district to increase segregation, many are now using it to increase integration.

>As this issue continues to gain national focus, I urge you to read about it and discuss it, keeping your Logo philosophy in mind.

If you would like to respond to these articles and commentary, or know of other Logo related articles and issues which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Ohio Logo Project Going Strong

"Integrating Logo into the Classroom" is the theme of a year-long project being undertaken by the Educational Computer Consortium of Ohio (ECCO) in 1985 - 86. With the help of a grant from the Martha Holden Jennings Foundation, several workshops and two Logo conferences are being held.

During this project, six of the hands-on workshops present Logo materials that may be taken back to the classroom. Topics includs elementary math and science, elementary language arts and social studies, secondary math and science, and secondary language arts. For those teachers who are looking to move further into the use of Logo in the curriculum, Michael Tempel of Logo Computer Systems, Inc., is presenting three workshops which explore the topics of language, math, and science projects with Logo.

The spring Logo conference of 1985 was the starting point and the 1986 spring conference will wrap up the project. ECCO has provided opportunities for the beginning Logo user and materials for teachers who do not have time to develop their own. The project is an exciting venture for all who participate.

For more information, write to Josie Drushal, Elementary Computer Resource Teacher, Wooster City Schools, Wooster, OH 44691.

Listful Thinking

by Glen Bull and Paula Cochran

Story Time

```
?NLX.STORY.STARTER

Once upon a time MRS. JONES was thinking about how to integrate what her
students were studying in GRAMMAR into their WRITING assignments. She was
casually leafing through the latest NLX, when suddenly . . .
```

A story starter, such as the one above, is supposed to get the creative juices flowing. It should give the reader / writer some ideas, and a basis for continuing the story in his own way. Naturally, we expect that the story above will have a happy ending. In this month's column we will discuss how to use Logo to develop story activities and story starters geared toward specific language arts goals.

Getting Started

The Logo-generated Christmas story below is one which we recently received from a local elementary class. The class is using Apple Logo II, which allows upper and lower case text. Some words in the story are randomly chosen (shown capitalized), making the story slightly different each time it is printed on the screen (or printer). Our thanks to Cheryl Wissick and her students in the Albemarle County Public Schools for sharing this story.

```
There were 14 BUSY elves at the North Pole.
SUSAN was wrapping the TRUCKS and
GEORGE was packing the COLD sleigh.
Then Santa found LAZY ROY sleeping behind
the boxes of BIKES. Santa was glad the other
CHEERFUL elves were hard workers, so he and
SALLY treated everyone to HOT CHOCOLATE.
```

The students who wrote this story worked as a group to think of key words for the story, such as the toys, the names of the elves, and the adjectives which describe them. The Logo story procedure is picking from lists of these words.

Talking about the lists of words caused students to review some of their recent work on parts of speech. This will be discussed in more detail below.

The students got excited about the stories for two reasons: they enjoyed seeing what funny and interesting combinations happened each time the story procedure was called, and they enjoyed seeing their own suggestions appear in the stories. They were encouraged to use the stories as story starters, writing their own ideas about what might have happened next. The results were so successful that they used them in a holiday program for their school.

Before we get into how such a story procedure and some enhancements might work, let's look at a very simple story starter procedure.

Start at the Very Beginning

Students of all ages seem to enjoy seeing their names and the names of their friends in stories. We will need a PERSON procedure, which uses PICK. (If you are using Version 1 of Terrapin Logo, you will also need to type in the ITEM and COUNT procedures found at the end of the column.)

```
TO PICK :LIST
OP ITEM 1 + (RANDOM COUNT :LIST) :LIST
END
```

PICK is a general-purpose tool widely used within the Logo community, and we have employed it in a number of our projects. PICK is the only tool necessary for a basic story starter, such as we described in the November, 1984 Listful Thinking column, "Count Your Blessings" (NLX 3 (3), pp. 7-10).

```
TO PERSON
OUTPUT PICK [ SALLY TOM GEORGE SUSAN ROY BEN ]
END
```

This procedure picks randomly from the list of names we provided:

```
?PRINT PERSON
GEORGE
```

It is easy to incorporate PERSON and other similar procedures into a story starter. Using PERSON as a model, we could add an ANIMAL procedure for a story about a class trip to the zoo.

```
TO ANIMAL
OUTPUT PICK [ GORILLA PARROT LION ELEPHANT MONKEY SEAL ]
END
```

```
TO ZOO.STORY
PRINT [ Mrs. Jones' fifth grade class went to the zoo. ]
PRINT (SE  PERSON [made] ANIMAL [noises on the bus.])
PRINT (SE  PERSON [was hoping to see a real] ANIMAL [.])
PRINT [ ]
PRINT (SE  [The class was walking by the] ANIMAL [cage])
PRINT (SE [when they heard a] ANIMAL [.])
PRINT (SE [Suddenly,] PERSON [looked up and saw . . .])
END
```

Depending on which words are PICKed, you might get the following story starter when ZOO.STORY is called:

```
Mrs. Jones' fifth grade class went to the zoo.
BEN made LION noises on the bus.
GEORGE was hoping to see a real GORILLA.

The class was walking by the ELEPHANT cage
when they heard a MONKEY.
Suddenly, SUSAN looked up and saw . . .
```

Any relevant class project or seasonal event can be used as the basis for a story starter. If only one story starter is needed, it is hard to see much advantage in using Logo for the task instead of a regular word processing program. The differences are apparent, however, when the teacher prints out 20 story starters, all with the same basic structure but each slightly different. Using this method, the teacher can easily print out a unique story starter for each student in the class.

Stories and Grammar

How can Mrs. Jones use the Logo story starter template to emphasize grammar goals, too? We strongly encourage teachers to involve the students in the design of the story starter, and not just use the finished product for a writing exercise. Depending on the age and Logo background of the students, this can be done on a group or individual basis.

The ideal situation is for a teacher to be working with a small group of students, using a large monitor or projector so that everyone is looking at the same screen. Then together they can work on making lists of words that will produce interesting stories.

Students do not necessarily need to know how the story starter procedure works, or even what it looks like. The teacher should prepare the skeleton of the story ahead of time, and get the lists of key words started with a *few* good examples. As the activity begins, the teacher explains the topic of the story, and shows the students some examples of the story she has made so far.

After they talk about the stories on the screen, she asks them to help her think of more words for the story. She edits the appropriate procedure (PERSON, ANIMAL, etc.) and adds their suggestions for words that match that category and that would sound good in the story.

```
TO ANIMAL
OP PICK [ GIRAFFE SNAKE _______ _______ _______ ]
END
```

They should go back frequently to look at whole stories on the screen, to see if the new words do fit in the story (and to see the interesting results!).

The story starter should be written to emphasize the parts of speech that the teacher wants the children to think about and discuss. For example, the Elf Story presented earlier emphasized nouns and adjectives. That is, the words which changed in the story and which the students discussed most were nouns (trucks, bikes, hot chocolate, etc.) and adjectives (cheerful, lazy, cold, busy, etc.).

The same story could have easily been used to work on verbs and adverbs. Even the names of the procedures can emphasize recent grammar lessons. With apologies to the authors, here is a possible variation of their Elf Story, revised to make it a lesson on verbs and adverbs.

```
TO ADVERB
OP PICK [ HAPPILY QUICKLY WILLINGLY NOISILY ]
END

TO TRANSITIVE.VERB
OP PICK [ PACKING WRAPPING WASHING FIXING TYING ]
END

TO INTRANSITIVE.VERB
OP PICK [ SLEEPING WORKING THINKING SNORING
RESTING CRYING LAUGHING ]
END
```

Using the new lists, here is what the Elf Story might look like:

```
There were 14 elves working HAPPILY
at the North Pole. SUSAN was
WRAPPING the gifts and
GEORGE was PACKING the sleigh.
Then Santa found ROY SLEEPING
behind the boxes of new bicycles.
Santa knew that the other elves had been
WORKING, so he and SALLY QUICKLY
treated everyone to milk and cookies.
```

The story starter procedure can be written like ZOO.STORY, using PRINT statements and SENTENCE to combine the elements of each phrase in the story. This is the most straightforward way to write the story template, but typing PRINT, etc., can become a nuisance if you use this and other text-heavy applications frequently. This can be remedied and YOU can enhance the simple story starter through the use of a few Logo text tools described below.

Text Tools

Last month, you recall, we outlined the concept of a list machine. A list machine is a basic template or model for a procedure which operates on a list. With a few slight changes, the basic list machine can be modified to perform a variety of other tasks.

INSPECT is a basic Model-T list processing machine that we have discussed in previous columns. A slight modification to INSPECT will change it into a story - printing machine.

```
TO INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

Listful Thinking continued

If you are using Terrapin Logo, use the following version of INSPECT.

```
TO INSPECT :LIST
IF :LIST = [ ] STOP
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

INSPECT takes a list of items, and prints each item on a separate line.

```
?INSPECT [A B C]
A
B
C
```

If we enter text into a procedure without putting in any PRINT statements, we can still use INSPECT to print the text on the screen later. For example, if we start a story like this:

```
TO SAMPLE.STORY.TXT
MRS. JONES' FIFTH GRADE CLASS WENT TO THE ZOO.
EVERYONE HAD A GOOD TIME.
END
```

we can then print the story like this:

```
?INSPECT TEXT "SAMPLE.STORY.TXT

MRS. JONES' FIFTH GRADE CLASS WENT TO THE ZOO.
EVERYONE HAD A GOOD TIME.
```

This eliminates the need to type in PRINT statements and brackets. (We have adopted the convention of adding the .TXT suffix to procedure names when the procedures consist of text and will not run correctly alone.)

Why can't we use INSPECT for our story starters, without modifying it? It is because we have procedures like PERSON, ANIMAL, etc., embedded in the text. ZOO.STORY contained the SENTENCE command and parentheses, in order to have the variable words and sentence pieces print out together. The first two lines of the ZOO.STORY procedure looked like this:

```
TO ZOO.STORY
PRINT [Mrs. Jones' fifth grade class went to the zoo.]
PRINT (SE PERSON [made] ANIMAL [noises on the bus.])
. . .
```

If we omit the PRINT statements, SENTENCE, and the parentheses, how will Logo know the difference between regular text and embedded procedure names?

Our Contest Entry

Two small changes in INSPECT produce a new procedure which can be used to print a story. (If we were not disqualified, this would be our entry in the NLX List Contest. See the end of the column for details about the contest, and for the Terrapin version of FORMAT.)

```
TO FORMAT :STORY
IF EMPTYP :STORY [PRINT [ ] STOP]
PRINT FORM.LINE FIRST :STORY
FORMAT BUTFIRST :STORY
END
```

(The boldface in the FORMAT procedure indicates the changes made from INSPECT. For comparision, here is the INSPECT procedure once more.)

```
TO INSPECT :LIST
IF EMPTYP :LIST [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

. The procedure FORM.LINE is placed after PRINT in INSPECT, and is used to form each line of our story. (For now, don't worry about how it does this. Just type it in.)

```
TO FORM.LINE :LIST
OUTPUT RUN FPUT "( FPUT "SE LPUT ") :LIST
END
```

Now we can leave out all the PRINT statements when we enter the text of our story. Create a procedure with the following text. (If your version of Logo does not permit use of lower case, enter the text in all capital letters.)

```
TO ELF.STORY.TXT
[There were 14 elves working] ADVERB
[at the North Pole.] PERSON [was]
TRANSITIVE.VERB [the gifts and]
PERSON [was] TRANSITIVE.VERB [the sleigh.]
[Then Santa found] PERSON INTRANSITIVE.VERB
[behind the boxes of new bicycles.]
[Santa knew that the other elves had been]
INTRANSITIVE.VERB [, so he and ] PERSON ADVERB
[treated everyone to milk and cookies.]
END
```

To print ELF.STORY.TXT on the screen, type:

```
?FORMAT BF TEXT "ELF.STORY.TXT
```

Obviously we don't want to bother typing this each time, so we made a procedure to do it:

```
TO DISPLAY :PROCEDURE
FORMAT BF TEXT :PROCEDURE
END
```

Once these procedures are in place, you need only type:

```
?DISPLAY "ELF.STORY.TXT
There were 14 elves working QUICKLY
at the North Pole. ROY was
PACKING the gifts and
GEORGE was WASHING the sleigh.
Then Santa found SUSAN RESTING
behind the boxes of new bicycles.
Santa knew that the other elves had been
WORKING, so he and BEN HAPPILY
treated everyone to milk and cookies.
```

This use of DISPLAY and FORMAT saves the hassle of many PRINT statements, etc., not only for the teacher, but also for students assigned to work independently on stories or story starters. They will be much less likely to be bothered with Logo syntax details such as missing parentheses. This allows them to focus their attention on the English goals of the exercise.

Listful Thinking continued

Of course, the tools should be provided by the teacher on disk, along with a model of the inside of a STORY.TXT procedure. The only thing students will need to know to make it work, besides how to edit a procedure, is to type DISPLAY "STORY.TXT.

How *Does* FORMAT Work?

Earlier we asked you not to worry about how FORMAT works. We just wanted you to type it in and use it. However, we know that there are a few of you who will worry anyway. If you are one of those persons, here's an explanation. (If you're not interested, skip ahead to the next section. You won't hurt our feelings!)

FORMAT is a straightforward variation of INSPECT. The primary difference is that the middle line (PRINT FIRST :LIST) becomes (PRINT FORM.LINE FIRST :LIST). The FORM.LINE procedure takes a line of the story such as this:

```
[ [There were 14 elves working] ADVERB ]
```

FORM.LINE inserts the command to form a sentence (SENTENCE, or SE for short) and places parentheses around the text. This turns the list into a valid line of Logo code.

```
[ ( SE [There were 14 elves working] ADVERB ) ]
```

FORM.LINE runs the code, by using the command RUN, to form a sentence. Lisp is the only other higher level computer language we know of that can do this.

Mixed Parts
(Parts is Parts)

Now that DISPLAY, FORMAT, and FORM.LINE have removed some of the overhead of story starters and stories which use word lists, let's look at a couple more ways to use them in grammatically oriented activities.

The class which sent us the original ELF.STORY did an interesting exercise using the story template alone. Instead of PERSON, ADVERB, etc., those places in the story were replaced with blanks. Each student filled in the blanks on a worksheet with his own words, and then compared them to the stories produced with the word list procedures.

Using the story with blanks could be either a pre-computer activity, to get everyone thinking before the group activity, or a follow-up activity to carry over the use of certain parts of speech in a writing exercise.

Another possibility for a follow-up activity involves scrambling the story a bit. Suppose Logo generates a story with key words in the wrong places? Naturally, the key words are verbs or adverbs, or another part of speech which the class is studying.

Let's scramble ELF.STORY.TXT, by just moving INTRANSITIVE.VERB and TRANSITIVE.VERB around:

```
TO ELF.STORY.TXT
[There were 14 elves working] ADVERB
[at the North Pole.] PERSON [was]
INTRANSITIVE.VERB [the gifts and] PERSON
[was] INTRANSITIVE.VERB [the sleigh.]
[Then Santa found] PERSON TRANSITIVE.VERB
[behind the boxes of new bicycles.]
[Santa knew that the other elves had been]
TRANSITIVE.VERB [, so he and ] PERSON ADVERB
[treated everyone to milk and cookies.]
END
```

Now when we print the story, it could come out like:

```
?DISPLAY "ELF.STORY.TXT
There were 14 elves working QUICKLY
at the North Pole. ROY was
SNORING the gifts and
GEORGE was THINKING the sleigh.
Then Santa found SUSAN TYING
behind the boxes of new bicycles.
Santa knew that the other elves had been
FIXING, so he and BEN HAPPILY
treated everyone to milk and cookies.
```

We could have scrambled any elements of the story, but the contrast between verbs which do and do not take objects would be a good place to start in some classes. Using a printout, each student should unscramble the story he received, using the same words in different places. Starting with familiar stories they have already thought about, students can work up to repairing the syntax in stories which are new and have more than one set of mixed parts.

The teacher can use class events and current topics of interest as the basis for the stories, reinforcing not only grammar in writing activities, but also new vocabulary from social studies or science class.

Good luck, Mrs. Jones, wherever you are!

The Contest!

A Koala Pad will be awarded for the best procedure which makes use of the fact that a procedure is just a series of lists. Submissions must be received by February 28, 1986. Send your procedure listings to Glen Bull and Paula Cochran, University of Virginia Curry School of Education, 109 New Cabell Hall, Charlottesville, VA 22903.

In addition, if you have other list processing activities you would like to share with us, please send them to the above address, or CompuServe 72477,1637.

Procedures for Terrapin Users

```
TO FORMAT :STORY
IF :LIST = [ ] PRINT [ ] STOP
PRINT FORM.LINE FIRST :STORY
FORMAT BF :STORY
END
```

Procedures for Terrapin, Version 1 Only

We use the Conan (hack and cut) approach to programming. Our COUNT and ITEM procedures published previously called other (recursive) procedures which did the work. Alison Birch wrote to point out that COUNT and ITEM can be (and typically are) written more efficiently by making full use of recursion within single procedures. These compact

Listful Thinking concluded

recursive versions follow. (Similar versions are found on the disk, Terrapin Utilities II, which can be obtained from Terrapin Inc., 222 Third Street, Cambridge, MA 02142, tel. (617) 492-8816.)

```
TO COUNT :OBJECT
IF EMPTY? :OBJECT OP 0
OP 1 + COUNT BF :OBJECT
END

TO ITEM :NUMBER :OBJECT
IF :NUMBER = 1 OP FIRST :OBJECT
OP ITEM :NUMBER - 1 BF :OBJECT
END

TO EMPTY? :OBJECT
OP ANYOF :OBJECT = [ ] :OBJECT = "
END
```

Alison also sent a sample issue of her newsletter, *Kaleidoscopes*, which we found to be excellent. For more information, write to Kaleidoscopes, Computers for a New Education, 57 Waverly Street, Suite 100, Belmont, MA 02178.

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. Paula Cochran is finishing her doctorate at the Curry School of Education at the University of Virginia. She is interested in Logo applications in language arts and speech-language pathology.

TurtleTips

by Jane Toth
and Donna Lanyi

Our *hearts* go out to students who are "going around in circles" trying to figure out the CIRCLE and ARC commands. After working with Logo for three years, we still struggle with choosing the correct numbers. We have finally figured out some basic steps that helped us conquer our "arcphobia."

The Turtle Teacher

One of the first concepts children have to understand is that circles and arcs on the computer are really a series of connected straight lines with turns in between. To demonstrate this idea, take your class to the gym and use the painted circles on the floor. Have all students use their own feet as straight line segments. Have them walk around the circles heel-to-toe to find out how many of their straight lines it takes to go around once. Discuss with your class why students' numbers differ (size of circle, length of straight segment, size of foot, etc.).

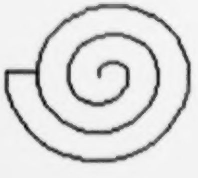
An effective manipulative device to illustrate this same idea is the use of straw sets. Create straw sets by cutting regular drinking straws to various lengths. For example, make a set of one-half inch long white straws, a set of one inch long striped straws, and so on. Have students connect a given number of straws of a particular set to create a circle. They could also construct straw circles by threading a string through a number of straws and then curling the string to form a circle. How big is the white circle? How about the striped one?

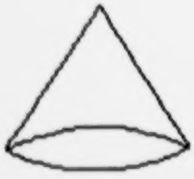
Next, have students place straws on given circle patterns. How many white ones were needed? How many striped ones? Again discuss the importance of the length of the straws and the amount of turn between each straw. (Ed. note: For additional related activities, see "Tipps for Teachers: Wonderful Wiggles," in the January 1985 NLX.)

Next, we give our students two separate worksheets, one for arcs and one for circles each with a set of given commands. Students type in the commands and copy the shapes from the monitor onto their worksheets. We have found that students keep these sheets as references for using circles and arcs in other procedures.

Ready for a challenge? Elizabeth Moody from Trinity School in New York gave us worksheets with the following shapes. She suggested giving students the first few commands and challenging them to finish the designs.

```
TO SNAIL
REPEAT 30 [ RT 6 FD 1 ]
REPEAT 30 [ RT 6 FD 2 ]
...
```



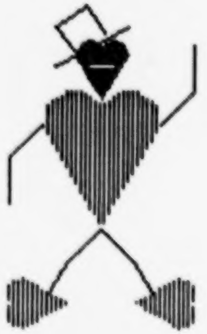


```
TO CONE
RT 60
REPEAT 30 [ FD 4 RT 2 ]
...
```

Procedure of the Month

How's your "heartwork?" Have students put their knowledge of arcs and circles to practical use. They could design heart people for bulletin boards, Valentine cards, heart mobiles, and Valentine stories with their heart people as characters. Students could even turn their stories into a Valentine puppet play using their heart people on popsicle sticks as the characters. If you are interested in a copy of our heart person procedure, please send us a self-addressed stamped envelope.

TurtleTips continued



We invite any teachers or elementary students to send Logo procedures or ideas to be shared in this NLX column. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691, or Compuserve 72477,2623.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher to Teacher

A Review of New Logo Materials

by James Fry

This month's column features a book which represents the next wave of Logo materials that are finally reaching the bookstores: books on teaching *with* Logo, not just teaching Logo.

Teaching with Logo - Building Blocks for Learning, by Molly and Dan Watt, 1986, published by Addison-Wesley Innovative Division, 2725 Sand Hill Road, Menlo Park, CA 94025, $16.88.

Target Audience: Anyone who teaches with Logo to students of any age.

Content: The book focuses on the learning of a group of people in a Logo community. The main focus is on a student named Lynn and how she works and responds to eight different sessions of working with Logo. *Teaching with Logo* is a unique demonstration of learning with Logo and the teaching of many "powerful ideas."

Format: The book consists of an Introduction and Guide, 9 chapters, an appendix, and an index. The introduction has an excellent discussion on Logo and and a list of 20 powerful ideas for those who use Logo. (Ed. note: See also Molly Watt's front page article in the September 1985 NLX.) Each chapter progresses to an increased level of sophistication and shows specific strategies that a teacher may use. The chapters also make comments on what has been taught and learned. Each chapter ends with examples of class records of the work done by each student.

The last two chapters follow a different pattern. The role of the teacher is discussed in more detail, and advanced Logo projects are offered.

The procedures in the book are written in Terrapin / Krell (MIT) versions of Logo. A listing of the procedures for other popular Logo versions is contained in the appendix.

Strengths: This is a book that many people need to read. A number of school districts are giving up on Logo because they have never really understood it. They have been teaching Logo and never realized that they should be teaching *with* Logo.

The conversations that have been recorded between students and teachers are valuable teaching aids. The notes and Logo class records at the end of each chapter offer many useful ideas.

Comments: Thank you, Molly and Dan! It is wonderful to see this exciting and unique guide with fresh Logo ideas. Keep up the good work!

• • • • •

What new Logo materials and books do you find useful in teaching? We invite you to send information to the Logo Information For Teachers (LIFT) group, PO Box 5396, Plymouth, MI 48170, or send electronic mail on CompuServe to 76317,565.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI, and is a co-founder of the Logo Information For Teachers (LIFT) group.

Logo Institute Announced

Dan and Molly Watt and Tony Stavely have scheduled their summer Logo institute to be held at Keene State College, Keene, NH, July 14 through 27, 1986, to follow the Logo 86 conference at MIT. Past participants have spoken highly of this institute (see Logo Teacher Feature on page 5). For more information, write to Tony Stavely, Logo Institute, Keene State College, Keene, NH 03431, or call (603) 352-1909, ext. 554.

Logo Disserts:
Dissertations Dealing wih Logo

by Barbara Elias

What are the programming abilities and thought processes of five year old children when they learn to program a computer with Logo? What problems do they encounter and how do they feel about programming? A study of five year olds by Sandra N. Hines may provide some insights for kindergarten teachers and curriculum planners.

Hines conducted a ten week study of children enrolled in a public school kindergarten class. Children were randomly selected from a group which had passed a screening test on numeral and letter recognition.

Daily programming sessions of about twenty minutes each were held in a room near the regular classroom. Games were used to introduce the computer and Logo to the children. Additional games served to help the children in orienting their bodies in space. Each child wore labels on the backs of his or her hands to reduce confusion about directionality.

Prior to programming on the computer, children drew their designs on paper. These designs were then enlarged on a drawing on the playground. Children walked along the lines, counting the number of steps needed for each part.

The researcher recorded the number of steps required; these numbers became the inputs for the computer programs. After several outdoor drawing experiences, the children could write programs to draw their pictures using only the paper drawings. A number line was used as a teaching tool for number inputs, number quantity, and comparison of numbers.

Data resulted from (a) dribble files, notes, and tape recordings of individual sessions, (b) test results from a number quantity test and scoring on nine Piagetian tasks, and (c) from an informal interview. Two tests on programming, given at the end of the study, provided additional data.

The children in this study exhibited a variety of abilities in programming. All were able to work out procedures with the researcher's assistance. Two could be considered independent programmers, one could not be considered a programmer, and two could write procedures but experienced difficulty in correcting errors.

The number of procedures completed by the participants varied, and indicated their interest and ability to attend to programming tasks. Those who completed the most procedures liked programming. Those who completed the fewest procedures preferred the games, and therefore spent less time on projects.

Task perseverance, interest level, and the use of thinking strategies instead of repetitive behavior appear to be factors in programming success. Some five year olds can program, but they need help in programming. Flexible instructional arrangements which provide this help and the use of prerequisite skill games and programs which require little typing may be good starting points for computer "literacy" at the kindergarten level.

>>>Hines, S. N. (1984). *A Qualitative Analysis of the Computer Programming Abilities and Thought Processes of Five Year Old Children* (Doctoral dissertation, North Texas State University, 1984). *Dissertation Abstracts International*, 45, 1983-A.

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Q and A

by Jim McCauley

Q. I understand that some computer languages are compiled, while others are interpreted. What exactly does this mean? Is Logo compiled or interpreted?

A. A *source* computer program is a text which can be read and understood (one hopes!) by human beings. Unfortunately, most computers don't "speak" the high level programming languages (such as Logo and Pascal) that human beings prefer to use. Each type of computer has its own unique language, called "machine code."

There are two ways of arranging for a computer to "understand" and run the text. First, the source text may be *compiled* into a sequence of machine code instructions by sending the text through a compiler program. This program attempts to translate the entire text all in one gulp, as it were.

If there are errors in the program, the compiler will either stop, print out one or many error messages, or go completely crazy, depending on how "smart" the compiler is. If there are no errors, the compiler program usually saves the new machine code version of the program on disk, and the original human-readable text is no longer needed (although it is good practice to save it as well).

One Line at a Time

The second method of translating a program text into machine code is to *interpret* it. Remember that the compiler translates the whole program all at once. An interpreter program, on the other hand, takes each line of text, converts it into machine code, runs it, and then goes to the next line.

Compilers and interpreters each have their advantages and disadvantages. Compilers tend to produce

Q and A continued

programs that run very fast, but the programs often take up a lot of memory. Also, the process of compiling a program can sometimes take many minutes. Interpreters are usually slower in operation, but they are much handier for interactive program development. The programmer doesn't have to wait for a program to compile.

The Best of Both

Systems programmers have tried to combine the best features of compilation and interpretation. One way is to make up a very small and fast interpreter (sometimes called a "virtual machine" because it simulates a computer that doesn't exist) and then compile the source text to run on the interpreter. This means that a computer program can be stored in a very compact form and be run at a higher speed than one would expect from an interpreter.

Most versions of Logo are interpreted, but some clever measures have been employed to speed them up and reduce memory requirements. One of these is called "tokenization." It is similar to the idea of a combined compiler / interpreter. In some versions of Logo, you may have noticed that a program runs a bit (oops, sorry!) more slowly the first time than it does subsequently. This is because there is some streamlining of the code going on during its first run. Newer versions of Logo do the streamlining before the program is ever run.

As of this writing, ExperLogo for the Macintosh is the only version of Logo for microcomputers that is compiled. If you have ever seen a demonstration of ExperLogo, you know that Logo can be *fast!* Coral Software's Object Logo (due out some time next summer) will also be a true compiler.

David Thornburg has often pointed out that it is important that compiled versions of Logo be developed if Logo is to be taken seriously as a computing language. On the other hand, the interpreted versions that are commonly available seem to be entirely appropriate for many classroom uses.

I hope this information helps clear up some of the mystery of the words *compiled* and *interpreted.*

• • • • •

Each month, I will answer selected questions in this column about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to: Jim McCauley, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a stamped self-addressed envelope. You may also send questions to me on CompuServe at 70014,1136.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

NLXual *Challenges*

by Robs Muir

Logo Collectibles

From this side of the geographic fence, I'm sorry to report that Logo is in jeopardy in California's schools. This is a growing and serious problem. (I hesitate to bring up our problems since Los Angeles' high temperature today is 82° and Milwaukee's high is 1°!)

California's problem is *not* the weather, however; California's problem is that Logo is about to become institutionalized in elementary schools. On the surface, that sounds great, right? Yet, Logo won't be institutionalized in the "right" way. Instead, we'll do to Logo what we did to New Math. The buzzwords out here are "*integrate* computers into the curriculum;" in other words, "...don't change, subsume!"

Logo's principal attraction (for me, anyway) was its outspoken advocacy of child centered educational environments in which children could take an active role in structuring their own knowledge. Logo's promise for a more humane classroom where children and adults could share in meaningful intellectual discoveries is about to be lost in our state's drive for a more "productive" educational industry. "Back to Basics" is a killer; it will surely kill the enthusiasm of our students for this new technology if we allow educational policymakers to dictate continuums, scope and sequence charts, and performance objectives by grade level without some sensitivity to the individual's learning needs.

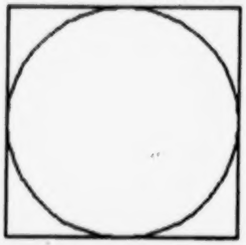
Sound familiar?

As a teacher, I need to justify to myself, my parents, my students, and my administrators that my classroom - with its noise, excitement, and Logo - is as effective a place as that stuff of principals' dreams: the quiet room where every child is on the same page and same paragraph. Logo teachers need to demonstrate that we *can* teach the content of the State Mathematics Framework while playing turtles.

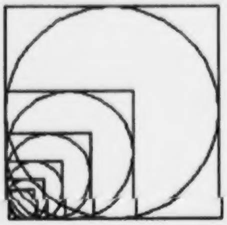
Let's keep this question (challenge?) in mind as you try to solve some of these oldies but goodies.

Focused Challenge # 1

Should my kids memorize the value for pi (π) from a textbook or solve a Logo challenge instead? Draw a square and inscribe a circle within it. (See NLXual Challenges, January 1984.) Can you solve this problem without resorting to π? Does this problem require you to "discover" and learn about π?

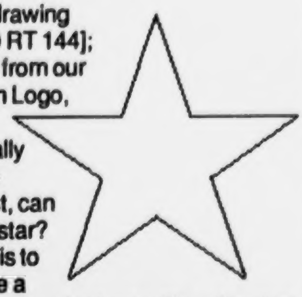


NLXual Challenges continued

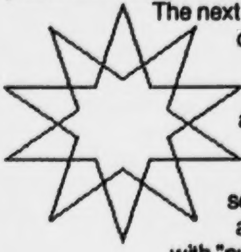


Generalize your solution so that you can draw a square of any size and the corresponding circle; you'll probably learn π more concretely than you did in your junior high school geometry class.

Focused Challenge # 2

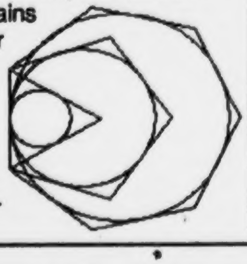


We all know the star drawing algorithm, REPEAT 5 [FD 50 RT 144]; that "lesson" was given to us from our teacher training workshops in Logo, or books that we used to learn Logo. Did any of us really discover the Total Turtle Trip Theorem for ourselves? First, can you create this variation of a star? The next task is to create a procedure (or two) with an input so you can draw this double star in any size. (See NLXual Challenges, December 1983.) I have yet to see this done without resorting to some trigonometric functions, such as sine and tangent. Can it be done with "pure" geometry, does anyone know?



Focused Challenge # 3

Ella Pfeiffer's POLYCIRC problem (See NLXual Challenges, November 1985.) remains a superb example of a challenge for high school (?) mathematics students which forces them to confront generalizations about simple polygons and circles. Draw a circle inscribed within an n - sided polygon. Here is a triangle, a pentagon, and a septagon.



Become Famous

The Logo community needs a collection of such simple focused problems which clearly address fundamental concepts. This could become a way of getting Logo legitimized by the establishment, since we can demonstrate how Logo can "teach" basic concepts. Notice that we can preserve the notion of discovery and invention through "problem solving," another hot buzzword.

Let's catalog some simple challenges that focus on specific pieces of knowledge, such as π and the basic trigonometric functions shown in the foregoing examples.

What Logo activities or projects have you done, with or without your students, that have successfully communicated a key mathematics concept or two? Send an outline of such an activity and we will create a compendium of Logo problems for others to "grab from." The key here is to isolate a specific concept that will lead a user to "discover" a new idea.

These ideas could be used by more traditional teachers to introduce or reinforce concepts, or they could become the core of a more comprehensive Logo-based math curriculum rich in problem solving.

What a gift this might be to the growing body of Logo literature! With a large community of Logophiles, we might create a massive collection of new activities. This could become the print equivalent of a samba school for Logo teachers.

Send your contribution to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, Ca 91711

Be sure to include a stamped self-addressed envelope if you would like a copy of the complete collection of challenges.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.

The National Logo Exchange, copyright © 1986 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223.

Editor...Tom Lough

THE NATIONAL LOGO EXCHANGE
The Logo Newsletter for Teachers

Volume 4 Number 7 FORWARD 100! March 1986 Page 1

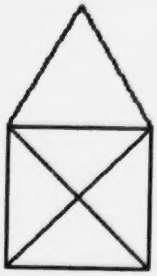
Euler and the Turtle

By Judi Harris

Let's face it, facilitators. Most of us have a Peter Pan or Wendy hiding behind our wise, adult facades, who reign in full glory whenever we sniff Logo in the classroom air! If "growing up" means losing the joyful, experimentative, problem-solving flexibility that is innately child-like, then we Logo-ers must reserve a special corner of our personalities that will never "mature." Chances are you have already done that, and your love for Logo stems partly from the opportunity that it affords to exercise the "never-neverland" option.

Modes of Mischief

Spring is in the air as I write at my lle, and memories flood back of how that special vernal aroma would incite even the most studious of my childhood friends to mischief! We perfected the technique (we thought) of appearing to listen in class, while really concentrating on something quite different.



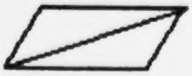
I remember specifically one puzzle "fad" that mesmerized us for several weeks. The challenge was to draw this figure without crossing a line already drawn, or lifting the pencil from the paper. Can you figure out how to do it given those constraints?

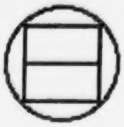
We finally did solve the problem, but the "magic" of it remained; some pictures could be drawn this way, and others couldn't. I doubt that any of us realized the inherent mathematical or problem-solving value of such an activity. We also would have been surprised to find out that one of the world's most prolific mathematicians pondered a similar puzzle in the early 18th century. He discovered the reason behind the "magic," also.

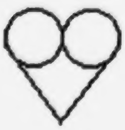
Euler and the Opus

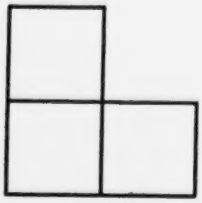
The Swiss mathematician Leonhard Euler (pronounced "oiler") had a Peter Pan in his personality long before Barrie conceived the character. Euler reportedly had 13 grandchildren, and is said to have created mathematical theorems with a baby on his lap and children playing at his feet. He wrote about 800 pages a year of good quality mathematical manuscripts, and is credited with concocting the original ideas of topology.

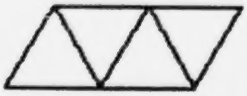
Enter the 20th century turtle, and the 18th century Euler line drawings like the one at left get a burst of Logo power! Here are some other designs which can be drawn without the turtle's pen going up or its crossing or retracing a line that it has already drawn.





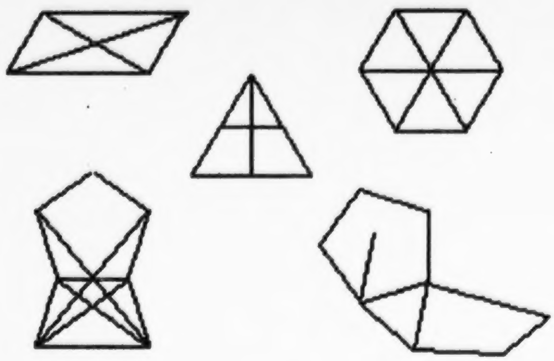






Euler continued

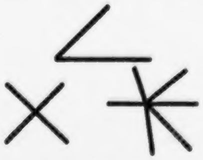
Designs such as these below cannot be produced according to those process specifications.



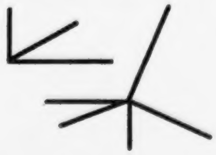
So what's the "magic?" What do the two groups each have in common within their individual attribute boundaries that is dissimilar across groups? Logo students could pose, consider, discuss, revise, prove, and disprove theories after sufficient experimentation with "turtled" Euler lines. Why don't YOU stop reading for now, and call out your Peter or Wendy to help you play with this?

Euler's Observations

Euler examined the line segments extending from the vertices of polygons and classified the intersections as "even vertices" and "odd vertices."



Even Vertices



Odd Vertices

He observed that the number of *odd* vertices in a figure is always even, and then went on to say:

- If there are no odd vertices in a polygon, then it can be drawn using any vertex as the starting point.
- If there are two odd vertices in a polygon, it can be drawn with Euler lines by starting at one odd vertex and finishing at the other.
- Otherwise, a polygon cannot be drawn without crossing or retracing lines, or lifting the pencil.

What powerful ideas! Do you want to play again, now, before you read on? Or did you already solve the Euler Logo puzzles?

See the Magic

Wise teachers capitalize upon children's natural interests and preferred activities. Why not make a set of polygons that can be drawn according to Euler's specifications? Challenge your students not only to solve the puzzles with the turtle, but to try to see the pattern to the "magic." Most importantly, they could make new Euler line puzzles for their classmates to solve!

Oh - by the way - if your Peter or Wendy wants to check how mine solved the Euler puzzles mentioned earlier in the article, help them to send a stamped self-addressed long envelope to:

Judi Harris
104 Woodside Road, B-306
Haverford, PA 19041

Judi Harris is an elementary school use facilitator in Haverford, PA, and a graduate education instructor for a number of institutes of higher learning. She is also an educational computing consultant whose clients include the Ministry of Education of Kuwait. Her CompuServe number is 75116,1207.

From The Editor

by Tom Lough

Educators are enjoined to encourage students to take risks. Taking risks is an important part of the Logo learning environment.

All too recently, we were reminded that the price to pay for taking a risk can be high. In spite of the shuttle tragedy, however, the quest for knowledge continues.

This space is dedicated to the memory of our professional colleague

Christa McAuliffe

and her fellow travelers,
Francis Scobee,
Michael Smith,
Judith Resnick,
Ronald McNair,
Ellison Onizuka,
and
Gregory Jarvis.

May their willingness to take risks be an inspiration to us all.

FD 104!

Tom

Tipps for Teachers

by Steve Tipps

Circular Quirks

Two weeks ago in an Introduction to Logo workshop, I loaded the startup file with Apple Logo by accident. In the startup file are the infamous CIRCLEL and CIRCLER procedures which draw nice circles. Of course, one or two of the groups found them and used them, while other groups worked steadily toward a circle solution with some frustration and many surprises.

When I asked the CIRCLER / CIRCLEL groups to create a circle using REPEAT, they looked at me as if I were crazed. Logo could already make a perfectly good circle without any effort or thinking on their part. Finally, if grudgingly, the preprogrammed circle groups began to use various REPEAT combinations with FORWARD and turns, but they were much less experimental in approach and more willing to accept the first circle they found.

Later in the day, they had projects which required understanding of circles and circle parts. The groups which had spent time playing with the circle were better equipped for the circular quirks found later. (Ed. note: See also "Circles, Arcs, and Headaches," by Sharon Burrowes, in the January 1985 NLX.)

Circles, Circles Everywhere

Experimenting with a circle reveals that a circle is decomposed into "move a little, turn a little, move a little, turn a little," etc. How much is a little? Many times students choose FD 1 RT 1. After some experimenting with REPEAT, they often find that REPEAT 360 [FD 1 RT 1] makes a good circle. I have learned to enforce the idea that a circle is a circle if the turtle starts and stops at the same position and heading. More than a circle or less than a circle is not a circle even if the figure is round. I do not refer to state transparency, but I do insist on the condition.

Having a group invent the REPEAT 360 [FD 1 RT 1] description of a circle is a great first step. An advantage of the description is the clear use of 360. However, stopping with the circle at this stage leaves the learner with a weak understanding of circle. A teacher who wants a full and powerful understanding of circle challenges students with leading questions. "Can you make the circle bigger or smaller?"

Another way to challenge this circle is to ask for a faster circle. The REPEAT 360 style is so slow that the reason for wanting to speed it up is obvious. Of course, you can hide the turtle to speed up the circle because Logo does not have to redraw the turtle 360 times. But other possibilities are more helpful in understanding the concept of circle.

Good Mistakes

Some learners begin by changing all three of the numbers at once.

```
REPEAT 75 [ FD 7 RT 3 ]  or  REPEAT 200 [ FD 6 LT 9 ]
```

These attempts are good mistakes on the path to circledom because they demonstrate clearly that the three variables interact by making larger, smaller, multiple, incomplete, faster, and slower circles in unanticipated ways. After a bit of good frustration, you might suggest going back to the idea of "move a little and turn a little" as a thinking model for controlling the circle. Instead of using REPEAT at first, try retyping the move and turn lines using <CTRL> and Y (LCSI) or <CTRL> and P (MIT) and follow what is happening. Start out with some patterns to see if you can make a circle with them.

Which of the following will make a circle?

```
FD 3 RT 3
BK 10 LT 10
FD 10 RT 45
FD 5 RT 15
BK -10 RT 60
FD 30 RT 30
```

Have the students add combinations of "move a little and turn a little" which make a circle and those which do not. You are hoping they find what conditions exist in all the circle combinations but not in the non-circle combinations. Learners should be able see that the turn a little must be 30 or less. A second relationship which you are trying to elicit is that the number of repetitions times the angles must be 360. Finally, the relationship of the number of repetitions times the "move a little" (or the total number of turtle steps around the outside of the circle) determines the approximate value of the circumference of the circle.

Tipps continued

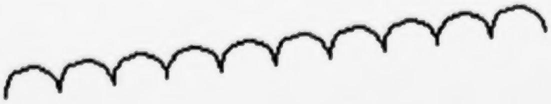
These are tough relationships to learn. Only through experimentation, which may take several hours on the computer over several days, will the learnings be solid. Easy circles with CIRCLEL or CIRCLER leave students with a weak understanding. But consistent work over time provides the background for students to use circles in many ways.

Making Waves

Sailboats are a natural project in Logo -- second only to houses in universal appeal. But what is a sailboat without water and what is water but blues waves across the bottom of the screen? Making waves is a project which employs circle parts. Recognizing that the wave is made of semicircles is easy. Getting the circles to behave properly is the hard part.

After students have a circle which is the right size (and speed), a semicircle is made by halving the number of REPEATs. One semicircle could be made from a circle described by REPEAT 24 [FD 3 RT 15]. Waves are semicircles with a turn of 180 between them.

```
TO SEMICIRCLE
REPEAT 12 [ FD 3 RT 15 ]
END

REPEAT 10 [ SEMICIRCLE RT 180 ]
```

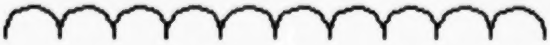


The waves scallop across the screen just as anticipated with one unanticipated variation. They slant up across the screen rather than being horizontal. Slanting waves are a puzzlement. Why don't they behave properly? The discrepancy can be illustrated by drawing half a hexagon or half an octagon.

```
REPEAT 3 [ FD 10 RT 60 ]
REPEAT 4 [ FD 10 RT 45 ]
```

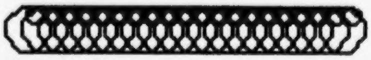
When half of a figure is drawn, the side does not return to the horizontal line where the drawing began. The same phenomenon is true with half a circle, though the difference is so small it is hard to see except when repeated several times. The diameter is on a slant, therefore the waves are on a slant. You can "cheat" on the semicircle by adding one extra FD to bring the wave back to level.

```
TO WAVE
REPEAT 12 [ FD 3 RT 15 ]
FD 3
END


TO WAVES
REPEAT 10 [ WAVE RT 180 ]
END
```



Loop-De-Loop

Making a loop-de-loop is another intriguing project using circle parts. The first problem is decomposing the loops into a large semicircle and a small semicircle. Slanting occurs with the loops as in the waves. If you "cheat" on the FD , the loops remain horizontal, although not centered.



```
TO LOOP1
REPEAT 12 [ FD 5 RT 30 ]
FD 5
END


TO LOOP2
REPEAT 12 [FD 2 RT 30 ]
FD 2
END


TO LOOP.DE.LOOP
REPEAT 20 [ LOOP1 LOOP2 ]
END
```

Another variation is to add a turn to the LOOPS for CIRCLE.LOOPS. The results for left turns and right turns are very interesting and different.

```
TO CIRCLE.LOOP1
REPEAT 18 [ LOOP1 LOOP2 RT 20 ]
END

TO CIRCLE.LOOP2
REPEAT 18 [ LOOP1 LOOP2 LT 20]
END
```

Variable LOOP procedures add ease and flexibility to the looping.

Tipps continued

Circular Thinking

The workshop brought home to me again the issue of preprogrammed procedures in Logo (See "The Issue of Instant," in the December 1983 NLX). When is it permissible and desirable to provide procedures for students? When is detrimental and undesirable? The answer to the question depends on the goals of the lesson and the tools which are needed to arrive at the goals.

With a group of teachers and most students, the intent in Logo is to provide an experience in problem solving. The problem is how to make a circle. The solution is a circle. Moving from the problem to the solution is the learning. In the process, learners find out that they can control the turtle, make a circle in many different ways, and discover some relationships about angles and perimeters. The knowledge and skills gained prepare the learner for new projects using the circle and its quirks

Circular thinking in most situations is frowned upon. However, in Logo, thinking about circles is exciting and important. Robbing students of the opportunity of working out the powerful relationships represented in the circle circumvents the learning possible with Logo. Dazzle Draw, Mouse Paint, and MacPaint are great programs with many possibilities including the drawing of the circle; I use each of them to make drawings and pictures. But none of them give me the understanding of circle that is part of Logo, at the heart of Logo.

By the way, there are also a couple of ways you can "cheat" on the semicircles using a turn. One way is similar to the use of FD. The other way is much more subtle (and more elegant).

If you have additional ideas to share, please send them to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or on CompuServe at 76606,1623.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Study Available

Logo Studies: Observations, Anecdotes, Effects, 1982 - 1985, a book of readings documenting the St.Paul, MN, Logo program, is available to NLX readers for $2.25. Send your check, made out to St. Paul Public Schools, to Geraldine Kozberg, St. Paul Public Schools, 360 Colborne Street, St. Paul, MN 55102.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Mark Mariotti

Very few teachers have the chance to spend most of their time guiding children of all ages into Logo discoveries. Mark Mariotti of Arvada, Colorado, is able to observe the full spectrum of elementary students every day in his job as computer specialist at Stott Elementary School near Denver. In Jefferson County, a major commitment to Logo instruction has been made. Although Logo is used in all 77 elementary schools, only a few of the faculties, like Mark's, have chosen to use a lab setting with a specialized teacher. This gives Mark some unique opportunities to work with students over several years and to observe their development as they use the computer.

Special Opportunities

Recently, Mark took advantage of these opportunities when sixth graders shared with first graders their project of illustrating and printing out nursery rhymes. The sixth graders used or modified traditional nursery rhymes or created their own. The sixth graders were tremendously proud of their work, especially when the first graders were suitably impressed! The first graders were quite proud, too, when they could read some of the nursery rhymes on the screen. Mark uses older students as lab helpers whenever he can. What a great way for students to learn about each other and about themselves!

Mark has been using tiling activities adapted from the NLX in various ways at different grade levels. Kindergarten students are using shapes that Mark created, Instant Logo, and a stamping procedure to stamp different shapes all over the screen. Shape stamping gives them lots of experience with spatial relations but uses very few Logo commands. Work with patterns ties in closely with clasroom work in the *Math Their Way* program.

Turtle Tiling

In the upper grades, Mark has used tiling as a way to introduce the editor. He challenged fourth graders to tile the screen with no overlappping. The third graders have been using tiles to build a frame around the entire screen. For the second graders, Mark wrote a procedure that dots the screen with variable spacing. They use dot paper and connect dots to form certain letters, then transfer the letter to the computer screen. Mark finds that each of his experiences with tiling has been different but really interesting.

Teacher Feature continued

Fifth graders at Stott have been actively involved in a science unit on space exploration, the shuttle, and the solar system. The students have been divided into smaller groups to use the computer to present information on the planets. These projects will be shared with the second and third grades. Mark used box charting with the groups to plan their projects. This planning format really helped students see which procedures were needed and modeled good problem-solving strategies.

The Time Crunch

Like most of us, Mark has found that time is a central issue in using Logo. There are lots of ideas to tie Logo in with the traditional curriculum, but time in the school day is scarce. Many problem-solving activities take the students more time than teachers imagine. Students of all ages need time to think. How do we fit in all the good things and still allow the students to spend all the time they need on a particular problem? Developing teachers' technical knowledge takes time, too.

Computer instruction sometimes seems to be an add-on, but its great potential will help us work through those difficulties, given enough time.

Do you know a teacher who should be featured in this column? Please send recommendations to:

Rebecca L. Poplin
2421 Fain Street
Wichita Falls, Texas 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Logo 86 Announced

Logo 86, the third international Logo conference to be sponsored by the Massachusetts Institute of Technology, is scheduled for July 9 - 11, 1986. A variety of settings will be provided to facilitate the building of Logo networks, including major speakers, presentations, poster sessions, and special interest group meetings. Logo 86 will be preceded by two days of tutorials. Make plans now to attend!

A call for papers has been issued for presentations and poster sessions. Papers from students are especially encouraged. The deadline is March 1, 1986. For more information, contact Brian Harvey, Logo 86 Program Chairperson, MIT, E15-309, 20 Ames Street, Cambridge, MA 02139.

For general conference information, write to Joyce Tobias at the above address.

NLXionary

A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"Computer Criticism vs. Technocentric Thinking" by Seymour Papert, *Logo 85 Theoretical Papers*, MIT, July, 1985.

>Don't let the title or the first few paragraphs of this piece scare you off, as they did me. Words like 'technocentric' and 'egocentrism' leap from the page in a semantic dare. I'm no match if the house is full of kids, or if I'm tired. But a recent quick scan also revealed subheadings such as "Do not ask what Logo can do to people, but what people can do with Logo" and "Bank Street vs. Kent State" or "Logo: The 'Cabbage Patch Kid' of Computation?" Score one for the Enquirer.

>Papert maintains that asking questions like "Does Logo produce good learning?" or "Does Logo improve thinking skills" are fundamentally the same as asking "Does wood produce good houses" or "Do hammers and saws produce good furniture?" Logo critics (e.g., Roy Pea et al. at Bank Street College) tend to ignore people factors such as skill, design and aesthetics - indeed, the whole culture in which Logo exists. Instead, they gravitate towards a medical model of experimentation, one that works fine for testing the effects of a certain drug, for example. Logo in this context is treated simply as isolated technical knowledge, rather than as one element in a particular type of learning culture.

>He cites the science classroom of George Franz at the Computer School in New York as having a Logo culture in the truest sense ... and whose students appreciate "Logo and the computer in a way that seems beyond the comprehension of many educational psychologists: using the computer not as a 'thing in itself' that may or may not deliver benefits, but as a material that can be appropriated to do better whatever you are doing (and which will not do anything if you are not!)." He compares this school to another in which the more they use Logo, the more they become different. They have different educational goals and use Logo for very different reasons.

>Papert is a believer in systems. If he wasn't a mathemetician he'd probably be a family therapist. A behavior problem can be seen as response from within an individual that's part of a linear cause / effect chain. Or it can be seen as one part of a complex, circular interaction that can be simultaneously both cause and result, helpful and hurtful, as it becomes intricately woven into the fabric of the daily lives of the people involved.

NLXionary continued

>At first glance, treatment strategies by one therapist might look the same as those employed by a skilled family therapist. ("Every night that the bed is dry, give him a sticker.") However, the family therapist thinks beyond this simple behavioral strategy and considers the systems in effect. Which parent should give the sticker? How can the grandparents' intrusion be handled? If the wetting is solved, how can the parents be helped with their sexual problem, which the wetting is helpfully distracting them from facing? These are but a few of the issues a systems therapist might consider when treating a problem.

>Likewise, Logo as treatment for education's problems (for example, lack of thinking skills) must be approached in a vein similar to a systems therapist - with the utmost regard to the culture, context, and patterns of interaction in which it is taught and used.

If you would like to respond to this commentary, or know of other Logo related articles and issues which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School. He and his wife, Robbie, are celebrating the arrival of Gillian Bree Wigley, a beautiful daughter born on January 20, 1986.

Logo Disserts:
Dissertations Dealing with Logo

by Barbara Elias

Logo - A Didactic Means of Re-education, by Nicole Michaud, presents some educational applications of Logo as a tool for the individualization of learning and as an invironment which can provide experiences for children in temporal, spatial, and causal relationships and concepts. Written in French, the study presents a description of observed behaviors and attitudes of children which evolved as they learned to program. Also included is a case study of an eleven year old boy and his progress in dealing with the problems he experienced with bugs.

Logo is viewed as an environment rich in opportunities for temporal, spatial, and causal experiences. For example, Logo incorporates operators of change, procedures represent temporal sequences and processes, and bugs present challenges with multiple possibilities for accomplishing desired purposes and discovery of new learning.

Children engaged in programming , over time, were observed initially to use a few simple procedures and little or no planning. They subsequently developed the ability to plan and organize projects using procedures and subprocedures, and to work with little support from the teacher. In a similar manner, typical initial attitudes *regarding* bugs caused a number of denial behaviors -

(a) hiding the bug by erasing it from the screen,
(b) ignoring the bug,
(c) blaming the bug on the teacher or the computer, and
(d) equating errors with personal failure and incompetence.

Over time, these children were able to use debugging techniques systematically and analyze bugs using physical movement, paper / pencil solutions, and organized searches of procedures.

The case study represents a description and chronology of the Logo learning experiences of an eleven-year-old boy. The boy is described as having learning problems in reading and mathematics, and difficulty in concentration and dealing with failure. He is able to grasp quickly the basics of Logo, but requires help with reading error messages.

After two months of two Logo sessions per week, he still refused to deal with bugs, and showed little planning of his projects. As time passed, however, he began to verbalize his difficulties with errors and to use limited debugging. An individual contract with the teacher was the turning point in his experiences. It assisted him in completing his projects and in ultimately dealing with errors.

At the end of another twenty sessions, he was more relaxed and was able to view bugs with humor and himself as something other than an absolute failure. Programming became an enjoyable experience and he took great pride in his accomplishments. The author notes the limitations of this study but feels that it offers many avenues for future research in learning with Logo.

>>>Michaud, Nicole. *Logo - A Didactic Means of Re-education* (Master of Education Thesis, University of Montreal, December, 1980).

Special thanks to Dr. L. Elie and his students, D. Elder, J. Pratt, M. Scales, and M. Villars, for the translation of this thesis from French into English.

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Listful Thinking

by Glen Bull and Paula Cochran

Summing Up

You can fool some of the people some of the time, but you can't fool the sum of the people all of the time.

A. Lincoln

Most of our "Listful Thinking" columns focus on lists of words, and what you can do with them, such as poetry, conversations, and stories. List operations can be applied equally well to lists of numbers. It's just that by training and temperament we are better qualified to discuss applications in the language arts ... and besides there are plenty of other people willing to talk about computers and numbers.

However, there are many list applications for numbers which are easy and fun. They can even be used in the social sciences, believe it or not. This month we are going to take a detour, and take a quick look at lists and numbers.

Number Sums

Everyone knows that Logo can add numbers.

```
?PRINT 2 + 3 + 7
12
```

In many versions of Logo, the same result can be obtained using the following command format:

```
?PRINT (SUM 2 3 7)
12
```

Terrapin Logo does not have a built-in procedure called SUM. However, don't worry. In just a moment we will give you a short procedure called ADD that does the same thing.

Why would you ever want to use SUM instead of a plus sign? Well, suppose that we had a group of numbers we wanted to add:

```
3 5 7 2 6
```

One way to add the list of numbers would be to put plus signs in between each of the numbers:

```
3 + 5 + 7 + 2 + 6
```

However, if we can use SUM, we can put a single operator in front of the entire list of numbers, and skip typing the symbols in between.

```
(SUM 3 5 7 2 6)
```

Now We're on a Roll

How might we use SUM to work with lists of numbers? We might want to collect a series of numbers, and then add or average them. See Steve Tipps' April 1985 NLX column for several classroom ideas.

We have created an example to demonstrate how SUM might be used which involves a simulation. We created a procedure to simulate the roll of dice. We might use this procedure in a Logo game, or we might use it in a class on probability. The procedure looks like this.

```
TO ROLL
OP 1 + RANDOM 6
END
```

Every time that we type PRINT ROLL, Logo prints a number between 1 and 6, just as in the roll of a die.

```
?PRINT ROLL
2
?PRINT ROLL
5
```

A similar simulation and other projects for students can be found in *Nudges: Apple Logo Projects* and in *Nudges: IBM Logo Projects*, by Steve Tipps, Tim Riordon, and Glen Bull. These books of Logo projects are published by Holt, Rinehart and Winston.

If each of the numbers 1 through 6 came up an equal number of times, what would the average be? Well, the sum of the numbers 1 through 6 is 21.

```
?PRINT (SUM 1 2 3 4 5 6)
21
```

If you have six numbers which add up to 21, their average will be between 3 and 4.

```
?PRINT 21 / 6
3.5
```

This makes sense, because the middle of the number line is between 3 and 4.

1 2 3 | 4 5 6

What would we actually get if we throw our simulated dice five times?

```
?REPEAT 5 [ PRINT ROLL ]
2
5
6
3
4
```

To find the average, we need to sum the numbers, and divide by five.

```
?PRINT (SUM 2 5 6 3 4)
20
?PRINT 20 / 5
4
```

The average we get is close to, but not the same as, the number we expected. Would we get an average closer to 3.5 if we threw the dice more times? It's as easy to say "REPEAT 100" as it is to say "REPEAT 5." However, we don't want to retype all the numbers that are printed on the screen. Why not let Logo collect them for us?

Here's how to keep a record of the numbers. First, initialize a variable, which we'll call DATA.

```
?MAKE "DATA [ ]
```

DATA, as you may have recognized, starts out as a list that has nothing in it. Each time we throw the dice, we'll *put* the new number that we generate as the *last* entry in the list. (The command LPUT in ROLL.DATA stands for "last put".)

```
TO ROLL.DATA
MAKE "DATA LPUT ROLL :DATA
PRINT LAST :DATA
END
```

Let's try it out.

```
?ROLL.DATA
3
?PRINT :DATA
3
```

Let's throw the dice three more times, and add these numbers to the end of the list.

```
?REPEAT 3 [ ROLL.DATA ]
2
5
1
?PRINT :DATA
3 2 5 1
```

Once we develop the right tools, it's as easy to create a list that has the results of 100 rolls of the dice as it is to create one with five numbers. Now that we have a list with four numbers in it (3 2 5 1), we need to add the numbers together. One way to do this is to use the command SUM. The question is how can we make SUM use our list of numbers? Here's a tool, called ADD, which will do the trick.

```
TO ADD :LIST
OP RUN FPUT "( FPUT "SUM LPUT ") :LIST
END
```

If you are using Terrapin Logo, it is necessary to create ADD in a slightly different way. Here's the short procedure that we promised you earlier in the column.

```
TO ADD :LIST
IF EMPTY? :LIST OP 0
OP (FIRST :LIST) + ADD BF :LIST
END
```

If you are using Version 1 of Terrapin Logo, you will also need the EMPTY? procedure.

```
TO EMPTY? :WORD.OR.LIST
OP ANYOF :WORD.OR.LIST = [ ] :WORD.OR.LIST = "
END
```

ADD adds a list of numbers:

```
?PRINT ADD [3 2 5 1]
11
```

Of course, since our numbers are already in the list :DATA, we don't have to type them out.

```
?PRINT ADD :DATA
11
```

Listful Thinking continued

To average the numbers in the list, add them and then divide by the number of items in the list.

```
TO AVERAGE :LIST
OP (ADD :LIST) / (COUNT :LIST)
END
```

If you are using Version 1 of Terrapin Logo, you will also need to type in a COUNT procedure before you use AVERAGE.

```
TO COUNT :WORD.OR.LIST
IF EMPTY? :WORD.OR.LIST OP 0
OP 1 + COUNT BF :WORD.OR.LIST
END
```

Now we can collect data in a list, and average it without retyping all the numbers.

```
?PRINT AVERAGE :DATA
2.75
```

You'll note that 2.75 is pretty far away from our expected average of 3.5 ... but of course we only threw the dice four times. Is there any relationship between how close we get to the expected value of 3.5 and the number of times we throw the dice?

Let's find out. First start out by creating an empty list again.

```
?MAKE "DATA [ ]
```

Then roll the dice 100 times, and add each number you get to the end of the data list.

```
?REPEAT 100 [ ROLL.DATA ]
.
.
.
?PRINT AVERAGE :DATA
3.61
```

Here's what we got when we rolled the dice ten times, and calculated the average. (We did this four times.)

2.8 3.2 3.7 4.0

Then we rolled the dice 100 times, and calculated the average. Here's what we got the first four times we did this.

3.4 3.4 3.5 3.6

Rolling the dice 100 times usually gave us an average closer to 3.5 (the expected number) than just rolling the dice ten times. We thought we might get even closer if we rolled our simulated dice 1000 times, but we didn't want to stay up all night. If any readers would like to do this experiment, we would be interested in the results.

What does all this prove? Well, a gambler may get lucky a few times, but in the long run Las Vegas can count on the house advantage. In a more scientific vein, it also demonstrates some of the principles of sampling error. For example, how many people do you have to ask to find out who is going to be the next president?

We had fun doing this, and we'd like to encourage you to give it a try. However, we recognize that this play with numbers may leave some people cold (especially in March). Let's see what all this might have to do with social studies.

Perceive It or Not

Frequently, public policy is determined as much by perception as by fact. It often doesn't really matter what the facts are, it only matters what people *believe* they are. (Even if the preceding sentence isn't true, it doesn't matter as long as you *perceive* it to be true.)

You can conduct a poll about the perceptions of your own class. For example, how old do you suppose fourth-graders think the president is? (We don't recommend that you ask them how old they think *you* are!)

The following COLLECT.DATA procedure will allow you to poll your class, and find out the average of their guesses. Of course, some of your brighter students might ask whether the range of the numbers might be a better measure than the average. Or possibly the guess that occurs most often might be more interesting.

```
TO COLLECT.DATA
PRINT [ AFTER THE FINAL GUESS, ]
PRINT [ ENTER "Q" TO QUIT, ]
PRINT [ AND PRESS RETURN. ]
PRINT [ ]
PRINT [ AFTER EACH QUESTION, ]
PRINT [ ENTER A NUMBER, AND ]
PRINT [ THEN PRESS RETURN. ]
MAKE "DATA [ ]
COLLECT.LOOP
END

TO COLLECT.LOOP
PR [HOW OLD DO YOU THINK THE PRESIDENT IS?]
MAKE "NUMBER FIRST READLIST
IF NOT NUMBERP :NUMBER [ PRINT.AVG STOP ]
 MAKE "DATA LPUT :NUMBER :DATA
COLLECT.LOOP
END
```

Listful Thinking continued

```
TO PRINT.AVG
PRINT [ ]
PRINT [ THE AVERAGE OF ALL THE GUESSES IS: ]
PRINT AVERAGE :DATA
END
```

If you are using Terrapin Logo, you will need to use the following version of COLLECT.LOOP.

```
TO COLLECT.LOOP
PRINT [ HOW OLD DO YOU THINK THE
     PRESIDENT IS? ]
MAKE "NUMBER FIRST REQUEST
IF NOT NUMBER? :NUMBER
    PRINT.AVG STOP
MAKE "DATA LPUT :NUMBER :DATA
COLLECT.LOOP
END
```

Here's how the COLLECT.DATA PROCEDURE works.

```
?COLLECT.DATA
AFTER THE FINAL GUESS,
ENTER "Q" TO QUIT,
AND PRESS RETURN.
AFTER EACH QUESTION,
ENTER A NUMBER, AND
THEN PRESS RETURN.


HOW OLD DO YOU THINK THE PRESIDENT IS?

73
HOW OLD DO YOU THINK THE PRESIDENT IS?

78
HOW OLD DO YOU THINK THE PRESIDENT IS?

74
HOW OLD DO YOU THINK THE PRESIDENT IS?

Q
THE AVERAGE OF ALL THE GUESSES IS:
75
```

By changing the question in COLLECT.LOOP, you can use these procedures not only to conduct a poll about the perceived age of the president, but about any question that comes up in class. Of course, after the poll, the fun part is finding out what the answer really is!

(HAVING) SUM FUN

Polls and averages are used to get at many social issues. To take a mundane example, why are the chairs in your class the size that they are? Does the average height of a chair reflect the average height of a child? Of course, no matter what size chairs are, they will be too large for some children, and too small for others (as Goldilocks discovered). The question is how to minimize mismatches.

This is the heart of democracy, and comes up in a good many issues that go beyond the size of classroom chairs. However, it is apparent that even a simple issue such as this can be used to illustrate principles ranging from ergonomics to the democratic process. We'll be talking more about this next month.

In the meantime, some of you may have noticed the quotation at the beginning of the column. This quotation is from Artemus Lincoln, who teaches fifth grade at a nearby school...

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. His CompuServe ID is 72477,1637. Paula Cochran is finishing her doctorate at UVA's Curry School of Education. She is interested in Logo applications in language arts and speech-language pathology.

LogoWriter Announced

Logo Computer Systems Inc. has announced a new Logo product called *LogoWriter* in a series of advertisements in major computing magazines recently. *LogoWriter* is a combination of the LCSI Logo language and a programmable word processor, all in one application.

Using a piece of paper as a metaphor, *LogoWriter* permits both text and graphics (with up to four turtles) on the front side of the "paper" and the ability to define procedures on the back. The computer screen displays either the front or the back side of the "paper."

An expecially interesting aspect of *LogoWriter* is LCSI's site license provision - permission to make unlimited copies! A school site can purchase *LogoWriter* for $395, and receive versions for the Apple II computers *and* the IBM PC family of personal computers. Also included are activity cards, project books, and a teacher's manual. In addition to being able to provide *LogoWriter* copies to each student, the school site can purchase an extension of the license to permit their students to take their copies of *LogoWriter* home with them, providing that all copies are returned to the school site at the end of the term.

For a free information packet on *LogoWriter*, call 1-800-321-5646. New York residents call (212) 265-5646. Or write to Logo Computer Systems Inc., 555 West 57th Street, Suite 1236, New York, NY 10019.

TurtleTips

*by Jane Toth
and Donna Lanyi*

We've found that our students are "chomping at the bit" for some new challenges after many months of practicing primitive commands, the REPEAT statement, and circles and arcs. Now is the time to introduce procedure writing.

The Turtle Teacher

Students love to save their drawings to share with others. To do this, they need to write procedures in most cases. When writing procedures, students also need to become familiar with editing. We've found that our students readily grasp writing procedures but become boggled with editing. We know of only one solution: *practice!*

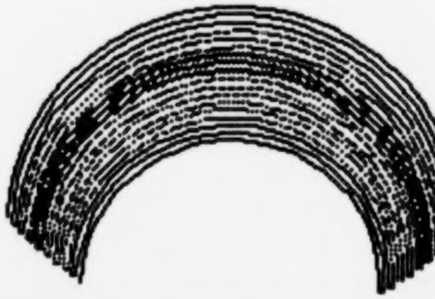
After introducing the concepts of procedures and editing to the class, we offer them plenty of opportunities for practice. We reinforce procedure writing with individualized packets of activities. Each packet contains a series of procedures that students can type in as models for correct procedural form.

Next, the beginnings of procedures that accompany a picture are given. Students are asked to complete the procedure. Of course, students are encouraged to try some of their own procedures that they have recorded in their computer notebooks.

We find that many students have difficulty learning the editing keystrokes. We've had our classes play EDIT (like BINGO) to help reinforce these concepts. Students write the CONTROL letters on their EDIT cards. The caller then says aloud the meanings of the editing keystrokes. Many similar games can be devised to help students tune these skills.

Once students become familiar with procedure writing and editing, they feel like they've "earned their wings." They are ready to be off and flying with their own ideas now.

Procedure of the Month



The pot of gold at the end of the rainbow can be your students' own creations. Take that pot of gold and utilize their ideas. Design a class display or newsletter featuring their projects to share with the school. If you are interested in a copy of a listing of our rainbow procedure, please send us a stamped self-addressed envelope.

Tell us about the interesting things you are doing with Logo in your school so we can share them with other NLX readers. Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691, or CompuServe 72477,2623.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

LIFT's

Teacher To Teacher
A Review of New Logo Materials

by James Fry

This month's column has reviews on two different types of Logo publications. They should be of interest to both the beginning Logo user and teacher and to the advanced user or teacher looking for extended activities with Logo.

For the Beginning User or Teacher

Logo Lessons Volume I - A guided curriculum for educators developing math and computer skills, by John Glantz, published by Logo Lessons, PO Box 39, Liberty, PA 16930, $29.95 (2 or more $26.50 each), plus $2.00 shipping and handling per manual. Returnable for a refund after 30 days if not satisfied.

Target Audience: Second grade to ninth grade teachers

Content: *Logo Lessons* is a guided manual for teachers who are looking for some direction and structure to help them teach Logo to their students. It covers the theory of Logo and a variety of off- and on-computer activities. *Logo Lessons* contains eight major lessons that the author states have been tested with second through ninth grade students. The teacher using these lessons gives one hour of lecture per week and has the students work either in class or during study or free time. The time required by fifth to seventh grade classes was about one month per lesson. The lessons are designed to build the students' abilities in computer literacy, math, and problem solving techniques. The lessons extend from the discovery of the basic turtle commands to multiple variables and repeats, along with team projects and advanced problem solving.

Teacher to Teacher continued

Format: *Logo Lessons* is set up in eight lessons with the same sequence for students and teachers to follow in each. A lesson requires between six and twelve hours to complete, depending on student age, computer use, and activities. Each lesson is organized around the following headings: materials for lesson, related topics, Logo topics, Logo concepts, classroom activities, computer activities, debugging and computer notes, bulletin board terms, thoughts about the lesson, mathematical concepts, activity centers, more Logo activities, and family oriented homework.

Logo Lessons also contains some "worksheets" that can be reproduced for classroom use. These are graphing sheets and some class and student note taking and organizational sheets. Video tapes and a training manual for each lesson will also be available to assist in teacher training in Logo.

Strengths: One thing struck me immediately: if you don't like *Logo Lessons*, you can return it within 30 days for a full refund. I would say that John Glantz has confidence in his work! The strenghts of *Logo Lessons* are the variety of activities, with the on- and off-computer work for students. The structure that it has should also be a big help to beginning Logo teachers or teachers who have not really found a comfortable direction or sequence with the instruction of Logo. The availablity of video tapes and a teacher training manual for each of the lessons should also be very valuable.

Comments: There is also a Volume II (geometric shapes, gears, and density) available and a volume III in the works that will cover list processing and clocks. NLX readers interested in more information should write to John Glantz, PO Box 39, Liberty, PA 16930.

Extended Logo Activities

Seven Microworlds for the Commodore 64, by Patrick W. Thompson, published by Cosine, Inc., PO Box 2017, West Lafayette, IN 47906.

Target Audience: First grade to university level students, depending on the microworld.

Content: I was able to look at five of the seven microworlds that Patrick W. Thompson has put together. They were titled:

Circuits - a microworld for logical inference,
Num.aids - a number theoretic toolkit,
Glue.boxes a microworld for arithmetic,
Inside.Logo - a microworld for exploring the inner workings of Logo, and
Integers - a microworld for integers and introductory algebra.

The author states that each has been used with students and teachers for at least one year before being published. The author also states:

> "The genesis of each microworld has been that I've found a piece of mathematics that has been particularly troublesome for students to understand, so I wrote a program that they could use to "visualize the content" and wrote a set of problems that focused their attention on important structural features of the content. In other words, the driving force behind each microworld has been the issue of enhancing mathematical understanding in nontechnical and nontrivial ways."

After studying these microworlds, I believe that the author has accomplished this task.

Format: The microworlds are organized into small booklets of 5 to 10 pages each. Each booklet explains what the particular microworld has been designed to accomplish. Commands and their actions are given, along with examples, questions and discussions. They are short and concise and give students a good starting point for their explorations.

Strengths: It is refreshing to see some alternatives or extended activities for using Logo in the curriculum areas. I like the format of the short booklets on each of the microworlds.

Comments: Let's hope we see more of these types of microworlds being published. I really like small booklet format.

•••••

What new Logo materials and books do you find useful in teaching? We invite you to send information to the Logo Information For Teachers (LIFT) group, PO Box 5396, Plymouth, MI 48170, or send electronic mail on CompuServe to 76317,565.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI, and is a co-founder of the Logo Information For Teachers (LIFT) group.

Turtle Maze Book Available

Spizzirri Publishing Co., Inc., publishes a line of educational activity coloring books and mazes. Their book on turtles contains drawings of 29 different species of turtles and separate mazes based on the drawings. The price is right: 99¢. You will find many ideas for Logo projects and displays in this little book. For more information, write to Spizzirri Publishing Co., Inc., PO Box 664, Medinah, IL 60157, or call (312) 529-1181.

Q and A

by Jim McCauley

Q. I understand that Logo has "descended" from Lisp, the language used in artificial intelligence research. Why shouldn't we just start children in Lisp, rather than having them do Logo first?

A. This question occurred to the original Logo research team back in the mid-1960's when Logo was being designed. There are several very good reasons for having children start with Logo:

(1) Short programs are easier to read in Logo than similar programs in Lisp. Here is a pair of examples. Both are functions that accept two lists and return one list that contains the common elements of both. The first is in Lisp, the other in Logo.

```
(DEFUN UNION (LAMBDA (LST1 LST2)
    (COND ((NULL LST1) LST2)
     ( (MEMBER (CAR LST1) LST2) (UNION (CDR
LST1) LST2) )
     (CONS (CAR LST1) (UNION (CDR LST1) LST2)) )))
```

```
        TO UNION :LST1 :LST2
        IF EMPTYP :LST1 [OP :LST2]
        IF MEMBERP FIRST :LST1 :LST2 [OP UNION
BF :LST1 :LST2]
        OP FPUT FIRST :LST1 UNION BF :LST1 :LST2
        END
```

If you study these for a bit, you'll be able to map the Logo version onto the Lisp version, allowing for vocabulary differences and the heavy use of parentheses in Lisp. Actually, all those parentheses make very long Lisp programs *much easier* to read, but they are confusing and tedious for beginners. Also, the vocabulary of Logo is much closer to English.

(2) Besides being easier to read, most versions of Logo have much nicer facilities for beginners than do most versions of Lisp. The versions of Logo that are widely available have fairly powerful editors that are very easy to use, while most Lisps have outrageously powerful editors that can take quite a long time to learn. Also, I know of only one version of Lisp for microcomputers that includes turtle graphics.

(3) In all versions of Lisp, the only things that you can write are *functions* (what we call *operations* in Logo). A function is a program entity which always produces an output, as in this month's NLXual Challenges. For many of the things that we have children do first in Logo (like turtle graphics), outputs are potentially confusing and not very useful. Logo gives us the capacity to write procedures that don't produce outputs. These are called commands. There is some controversy in the Logo community about the wisdom of this decision, since it can cause confusion when you start doing complicated things with the language. Of course, one option you have when your students hit those limits is to transfer them to Lisp. However, there is a tremendous amount that can be done with Logo before those limits are reached.

(4) Finally, Lisp was a language designed for general research work by professionals in artificial intelligence. Logo is a language designed for use by beginners of any age. Logo is easy to start with and has a tremendous amount of "growth room" before any serious limitations are reached. Although Lisp is a much easier computer language to learn than most, there is a significant threshold which must be overcome in learning it. But you can start using Logo immediately because it was specifically designed for that purpose.

In summary, then, Logo and Lisp share many similarities. Because of its low threshold, Logo is best suited for beginners. The powerful Lisp language is capable of carrying out the more complex projects of experienced programmers. The Logo - Lisp sequence is being explored by a few school systems already.

.....

Each month, I will answer selected questions in this column about Logo programming techniques and the thinking which leads to programming. If you have a question, send it to: Jim McCauley, NLX Q and A, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a stamped self-addressed envelope. You may also send questions to me at 70014,1136 on CompuServe's electronic mail system.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

Logo on Television

Kentucky Educational Television (KET) has scheduled a rebroadcast of a segment of its "Educational Computing" series which features an interview with Joyce Tobias on Logo in the classroom and the quality of Logo teaching materials available. The segment, EC 315, can be viewed April 3 and 4 on public television. Consult your local station for broadcast time, or write KET, 600 Cooper Drive, Lexington, KY 40502 for more information. Telephone: (606) 233-3000. KET also has available a videocassette featuring all Logo broadcasts from the 1985-86 season.

Logo Notes

MACUL 86, a conference sponsored by the Michigan Association for Computer Users in Learning, will be held March 13 - 14 in Grand Rapics. MI. The program features a number of Logo presentations and related activities. This particular conference has received "high marks" for several years, and this year promises to be no exception. For more information, write to MACUL 86, PO Box 628, Westland, MI 48185, or call (313) 595-2493.

Discovery III: Training and Technology for the Disabled is a conference scheduled for March 10 - 12 in Milwaukee,WI. Listed on the program is a presentation entitled, "Using Logo and BASIC with Special Needs Children," by Deborah Jolly. For more information, write to Office of Continuing Education, University of Wisconsin - Stout, Menomonie, WI 54751, or call (800) 45-STOUT. (22-STOUT in WI)

The World Congress on Education and Technology will be held in Vancouver, British Columbia, May 22 - 25, 1986. The program lists presentations by nearly every top name in educational computing. For more information, write to Anne Marx, WCET, 1155 West 8th Ave., Vancouver, BC V6H 1C5 Canada, or call (604) 734-2721.

NECC 86, the National Educational Computing Conference, is scheduled for June 4 - 6 in Sàn Diego, CA. Several Logo activities are scheduled. For a conference brochure, write to Jim Adams, ACM, 11 West 42nd Street, New York, NY 10036, or call (212) 869-7440.

IBM announces the publication of a series of IBM Logo manuals which are now available. A teacher's manual ($55), student's manual ($42), and a program diskette ($11) may be ordered through your local IBM representative. *Logo Learner*, a special version of IBM Logo for the IBM PCjr computer, is also available.

Computeach is a computer curriculum for grades K - 6 developed by educators of the East Ramapo (NY) Central School District. Volume II of the series deals with Logo, and is being offered for sale to other educators. For more information, write to Diane Gess, East Ramapo Central School District, Office for Curriculum, 461 Viola Road, Spring Valley, NY 10977, or call (914) 356-4100, ext.292.

Chris Templar and her associates at Johnson Bible College in Knoxville, TN, have written a Commodore 64 BASIC program which converts Koala Pad picture files into a format which can be read by Commodore Logo. For a copy of the listing, send a stamped self-addressed envelope to Commodore Koala Listing, NLX, PO Box 5341, Charlottesville, VA 22905.

NLXual Challenges

by Robs Muir

It's a Plot!

> *If we desire to form individuals capable of inventive thought and of helping the society of tomorrow to achieve progress, then it is clear that an education which is an active discovery of reality is superior to one that consists merely in providing the young with ready-made wills to will with and ready-made truths to know with.*
>
> *Jean Piaget, 1975*

When you tire of reading too many Logo books filled with cute little cartoon turtles and friendly bunnies, when all the Logo articles you read are concerned with "integrating Logo into the classroom experience," when Logo seems like nothing more than another (yawn) programming language, then it is time to go back and study the "classics" — the great works of the Logo's founders.

As Brian Harvey recently wrote, "Whatever happened to the revolution?" Have all the radicals been enfranchised by the system? We used to believe that the classroom would be integrated into the Logo experience, yet now we hear a different slogan. Today, every third-grade classroom lucky enough to have a computer offers a token lesson in Logo — just to show that they're progressive. But after the students' twenty minutes a week are up, it's back to the worksheets and dittos. This is not what the Logo forefathers and foremothers intended!

I recently dredged up Bob Lawler's "Designing Computer-Based Microworlds" (Byte Magazine, August, 1982), and on rereading this masterpeice, I remembered why I loved Logo from the start. Logo isn't a language; it's something more. It's an artifact of a community in love with learning.

The concept of microworld gets lost in the modern tempest of computer literacy. This month's challenge calls on you to build a world — simple, yet populated with rich ideas. A world infrequently explored by Logo's founders; the functional equivalent of Atlantis.

Mathematical functions return a value. Logo procedures (or primitives) that OUTPUT a value are termed "functions". The Logo primitive, SIN, is a simple example. Observe these numbers:

```
PRINT SIN -150
-0.5
PRINT SIN -90
-1
PRINT SIN -30
-0.5
PRINT SIN 0
0
```

Challenges continued

```
PRINT SIN 30
0.5
PRINT SIN 90
1
PRINT SIN 150
0.5
```

We could build our own functions in Logo:

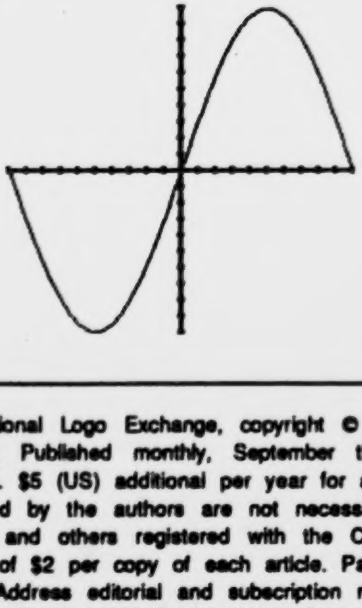
```
TO SQ :NUMBER
OUPUT :NUMBER * :NUMBER
END
```

or, for the mathematically literate, $f(x) = 3x^2 + 4$ is equivalent to the Logo notation,

```
TO FUNCTION :X
OP 3 * (SQ :X) + 4
END
```

The statement PRINT FUNCTION 3 will result in the number 31 being printed on the screen.

Using Logo's built-in Cartesian plotting tools, SETPOS (SETXY), SETX, SETY it is a simple matter to plot a function. All you have to do is run a range of X coordinates through your FUNCTION "machine" and have it return a set of Y coordinates for the turtle to move to. The SIN function above could look something like this:



PlotWorld

Your task is to build a general purpose plotting program that will allow you to plot any predefined function, such as — SIN, COS, or f(x)=-.75 * (sine (x * 3)). No fair looking at Lawler's previously cited article, or *Thinking with TLC Logo* (John Allen et al., Holt, Rinehart and Winston, 1984), Nancy Kovatch's "Functions of OUTPUT" (NLX, December, 1984), or *Apple Logo* (Drew Berentes, TAB Books,1984). There are others; resist any temptation and build your own solution.

Bob Lawler states "The programming needed to make a Logo plotting subsystem is nearly trivial, but this is precisely the virtue of a powerful language." A further virtue of any programming language is its flexibility; there are several different approaches you might take in designing this plotting microworld.

This microworld could be given to fifth-graders for exploration. Why should ideas powerful enough for the ancient Greeks be denied our students simply because this "isn't covered until junior high school?" The Logo revolution hoped to transcend tradition by building new tools to think and explore with. Viva la revolucion!

Send your solution, along with a self-addressed stamped envelope to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, CA 91711

You may also send solutions and suggestions for additional challenges to me on CompuServe at 70357,3403.

We will compile several interesting solutions and redistribute them to all contributors.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School.

The National Logo Exchange, copyright © 1985 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subecription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, Va 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223

Editor..Tom Lough

THE NATIONAL LOGO EXCHANGE
The Logo Newsletter for Teachers

Volume 4 Number 8 FORWARD 100! April 1986 Page 1

Procedure Inputs: Local Variables or Not?

by Sharon Burrowes

Our introductory high school computer science course, a prerequisite for further courses in computing, touches on the usual "computer literacy" topics and teaches Logo programming. Last fall, I found myself with a large number of students who had never seen Logo, but who had been exposed to BASIC in junior high school.

In this lively group, there was a lot of bantering about which language was "best," usually in good fun. Since Logo has local names defined by using procedure inputs or the primitive LOCAL and BASIC does not, I decided not to teach or allow the use of MAKE (for defining global names) until after we had worked with procedure inputs for a while.

A Local Problem

Shortly after we had begun work with inputs, I posed a problem to help students discover how to use procedure inputs for sending values from one procedure to another. The problem went something like this: Define a START procedure which accepts two numbers as inputs and then uses those two numbers in two other procedures to produce a product and a quotient. Both computation procedures must be called from the first procedure.

The next day, I came to class armed with my solution, which looked like this.

```
TO START :NUMBER1 :NUMBER2
FIRST.COMPUTATION :NUMBER1 :NUMBER2
SECOND. COMPUTATION :NUMBER1 :NUMBER2
END

TO FIRST.COMPUTATION :FIRST :SECOND
PRINT :FIRST * :SECOND
END

TO SECOND.COMPUTATION :TOP :BOTTOM
PRINT :TOP / :BOTTOM
END
```

The Student Solution

The bell had barely rung when Brad gloated, "I figured it out!"

"Ha!" I thought, but I said, "OK, let's have it." He dictated the following solution which I entered on the demonstration system for the class to see.

```
TO START :NUMBER1 :NUMBER2
FIRST.COMPUTATION
SECOND.COMPUTATION
END

TO FIRST.COMPUTATION
PRINT :NUMBER1 * :NUMBER2
END

TO SECOND.COMPUTATION
PRINT :NUMBER1 / :NUMBER2
END
```

He didn't have inputs for either of the computation procedures. Once the procedures were defined, I self-assuredly typed START, fully expecting Logo to respond with

```
NUMBER1 HAS NO VALUE IN FIRST.COMPUTATION:
PRINT :NUMBER1 * :NUMBER2
```

Local Variables continued

Much to my surprise, the procedures worked! Doug was chuckling, Curtis was laughing, and Brad was having the time of his life! All my emphasis on the local nature of procedure inputs came crashing down!

Back to the Drawing Board

After class, I checked every Logo reference that I had. Everything told me that procedure inputs *were* private or local to the procedure in which they were defined. After a few moments at the keyboard, I concluded that the guys were right. But, at that point, I didn't understand *why* their procedures worked. However, I soon solved the "mystery," thanks to some knowledgeable friends.

I found that a procedure input is local to the procedure in which it is defined *and* to all the subprocedures called by that procedure, as long as the calling procedure is running.

This manner in which Logo uses local names (called "dynamic scoping") can be used to avoid passing values from one procedure to another using additional procedure inputs. However, it complicates debugging since subprocedures become dependent on their superprocedures. It is a poor programming technique to use, in part because the modularity of Logo's procedures is lost. You can no longer debug each subprocedure separately.

Once you move beyond elementary programming in Logo, the scheme which you use to define names becomes an important concern. Here is a series of examples using a combination of both global and local names to demonstrate the kind of problems that can develop if you use Logo names in the way just described.

To begin, type

```
MAKE "X 10
```

and then define this procedure:

```
TO PROCEDURE.1
PRINT :X
END
```

If you then type PROCEDURE.1, Logo prints 10, since 10 is a global variable value. Similarly, if you define another procedure which calls PROCEDURE.1,

```
TO PROCEDURE.2
PRINT :X
PROCEDURE.1
END
```

and then type PROCEDURE.2, the result will be 10 and10, because the name X is global.

Change the Game

Next, make these modifications.

```
TO PROCEDURE.1 :X
PRINT :X
END

TO PROCEDURE.2
PRINT :X
PROCEDURE.1 :X
END
```

If you type PROCEDURE.1 15, then 15 is printed. If you type PROCEDURE.2, you may be surprised to find that 10 and 10 will be printed. Since no value is passed into PROCEDURE.2, it uses the *global* value of X (in this case, 10) and passes the 10 to PROCEDURE.1.

Now restore PROCEDURE.1 to its original form.

```
TO PROCEDURE.1
PRINT :X
END
```

If you type PROCEDURE.1, then 10 will be printed. Then modify PROCEDURE.2 as follows

```
TO PROCEDURE.2 :X
PRINT :X
PROCEDURE.1
END
```

and type PROCEDURE.2 20. Did you predict the result of 20 and 20?

This occurs because PROCEDURE.1 is a subprocedure of PROCEDURE.2 and PROCEDURE.2 has an input (which is local). This local name takes precedence over the global name X which has the value of 10.

The Local Wrapup

In summary, using the same name in many places can lead to confusion. Remember that:

1. Local names will be understood in all subprocedures of an active procedure.
2. The "most local" name always takes precedence. That is, if Logo encounters the same name, the value of the most local name will be used.

My experience with the scope of Logo names has made me more careful than ever about the scheme I use to create names in my procedures. It has also once again taught me the value of thoughtful and creative students. Who knows what they will teach me next?

Sharon Burrowes is the computer coordinator for the Wooster (OH) City Schools, and is currently serving as a member of the board of directors of the International Council for Computers in Education.

From The Editor

by Tom Lough

The Art of Intervention

Our son, Kyser, is well into his "terrible two's" now. But it is not all bad news. Like other youngsters his age, he is now beginning to attend to a task for more than fifteen seconds at a time! But, sometimes I'm afraid that I don't help his concentration very much.

Recently, he was involved with a jigsaw puzzle. I came into the room and immediately began to show him how to look for common colors in pieces that fit together. He would have none of it, of course! He had already established his own exploration routines. He didn't care if it took him longer to find the right pieces.

Later, I realized that I had intruded with no regard for what he was doing. My suggestion was made with no preparation or consideration of how my words would affect him. He resented my attempt to speed up his work. I resolved to do better next time.

The following week, he was working on the puzzle once more. This time, I approached quietly and spent a few moments observing what he was doing, trying to get into mental step with him. When I finally made a comment, it was with a better sense of timing and appropriateness. He smiled and we began to talk!

Still later, I thought about how I tried to apply the same idea to my students in their Logo work. Was I sensitive to what they were doing? Would I approach quietly and spend some time just observing? Did I try to get in step with them mentally before saying anything? Did I realize that sometimes it was not necessary to say anything? Would I restrain myself from giving solutions too readily in an attempt to speed up their work?

Effective intervention is an art. In order to be effective, intervention of any kind must be approached with sensitivity and empathetic understanding. I must always be award of how my intervention will affect the thoughts, the mood, the process of my students.

This is not an easy task, and I certainly have not fully mastered it. But I find myself improving little by little. My students (and Kyser, too) seem to appreciate it when I intervene effectively.

What experiences have you had with intervention in your Logo work? I'd be interested in learning about them.

FD 100! Tom

PS: Last month, we announced the availability of NLX brochures for distribution at conferences and in classes. We have filled many requests and even had to order a second printing! Thank you for your interest and support! Let us know if you need more!

Tipps for Teachers

by Steve Tipps

Random Redux

RANDOM is a Logo function which has many entertaining and enlightening possibilities. From its use as an input, to games, to tool making in Logo, RANDOM can be used to make random things happen. Some possibilities and off-computer activities were explored in a previous column "Random Thoughts" (NLX, March 1984), but the time has come to look at random again.

Randomly, Randomly

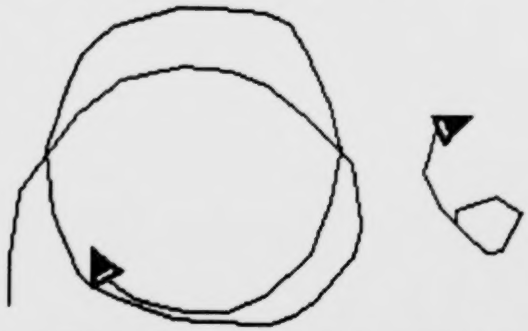
The idea of the turtle being able to pick out any number from 0 to 50 is easy to show with a PRINT command in a REPEAT statement.

```
REPEAT 20 [ PRINT RANDOM 50]
21
17
44
...
```

Students can change the number 50 to another number and compare the highest value and the lowest value printed. Several trials should confirm the lowest and the highest results from an input value to RANDOM.

RANDOM can also be explored with distance and direction commands.

```
REPEAT 50 [ FD RANDOM 30 RT RANDOM 30]
REPEAT 10 [ FD RANDOM 20 RT RANDOM 90]
```



These two examples draw continually right turning figures. For one project, a student wanted a random turtle movement which was sometimes a little to the left and sometimes a little to the right. The purpose was to simulate a mouse running. This was an excellent problem with a "correct" programming solution using an offset which I was wise enough not to tell.

Instead, the student worked with small numbers hoping that the turns would not take the inevitable curve to the right. Then came a moment of inspiration.

Tipps continued

The logic was explained to me like this:

"If RIGHT 20 LEFT 20 means the turtle is still going the same direction, then RIGHT RANDOM 20 LEFT RANDOM 20 would turn the turtle a little to the right then a little to the left. The difference between the turns would be the total turn depending on the size of the RANDOMs."

The invention led to a WOBBLE procedure which let the turtle move in a random, but approximately linear, fashion.

```
TO WOBBLE
LEFT RANDOM 20 RIGHT RANDOM 20
FORWARD RANDOM 20
END
```

Invention is wonderful learning, and this experience is one of many which alerts me to the value of Logo as a learning environment. I could have provided the approved answer with a long explanation about offsetting the values by adding 10 and subtracting random 20. However, the solution was just as effective and more intuitive.

Pickers and Choosers

Another use of RANDOM involves picking items from a list. The fundamental idea is to create a tool which can PICK any item from a list.

```
TO PICK :LIST
OUTPUT ITEM (RANDOM 1 + COUNT :LIST) :LIST
END
```

Understanding how PICK works may be interesting to some students, but the more important idea is to use PICK to do things. In their NLX column, "Listful Thinking," Glen Bull and Paula Cochran have used this procedure extensively in creating sentences and poems and stories. For example, PICK from lists of nouns and verbs for a sentence.

```
TO NOUN
OP [ BOYS GIRLS MONKEYS TRUCKS TEACHERS ]
END

TO VERB
OP [ JUMP RUN TALK ]
END

PRINT SE PICK NOUN PICK VERB
MONKEYS RUN
```

Other things can also be PICKed. A simple helper's list is one way to use PICK in the classroom. Use a procedure to output the list of students in the class.

```
TO STUDENT
OP [ STACY TRACY JAMIE SHAWN KEVIN MELISSA ]
END

PICK STUDENT
SHAWN
```

Random Fortunes

Another way to use PICK is as a fortune teller. Students are always interested in what will happen to them. Creating a fortune teller is a variation on the NOSY program shown in the January column. Students develop lists of possibilities to be used in a randomly generated future. Examples include future occupations, future places to live, or future transportation means.

```
TO OCCUPATION
OUTPUT [ TEACHER SCIENTIST [MARINE BIOLOGIST ] ]
END

TO PLACE
OUTPUT [ VULCAN SEATTLE [ ORION COLONY NINE ] ]
END

TO TRANSPORT
OUTPUT [ MASERATI ENTERPRISE PEQUOD ]
END

TO TELL.FUTURE
PRINT SE [ YOU WILL BE A ] PICK OCCUPATION
PRINT SE [ YOU WILL LIVE IN ] PICK PLACE
PRINT SE [ YOU WILL DRIVE A ] PICK TRANSPORT
END
```

The resulting futures should be entertaining and lead to extensions of many kinds. Creating lists in procedures rather than as global variables make them easier to edit and eliminates many tacky issues of where to use quote and colons.

In Jeopardy

Another use of PICK would be to create games. An easy example is based on the television game show "Jeopardy." Each procedure can be a category with a list of answers.

```
     TO PRESIDENTS
     OUTPUT [ WASHINGTON [16TH PRESIDENT ]
[LEAGUE OF NATIONS ] HERMITAGE [ FOUR TERMS ]
PEANUTS [ MARRIED TO DOLLY ] ]
     END

     TO STATES
     OUTPUT [ LARGEST [ LAST TO BE ADMITTED
TO UNION ] [EMPIRE STATE ] [SHAPED LIKE MITTEN]
[EPCOT CENTER ] [GRAND CANYON ] ]
     END
```

Tippe continued

```
PICK PRESIDENTS
PEANUTS

PICK STATES
SHAPED LIKE MITTEN
```

Categories can be invented to go along with any classroom units or interests. Of course, the inventors must keep track of the correct questions to go along with answers. A problem is likely to arise because the same item may be picked several times in a game. Rather than solving the problem for students, let them come to some resolution. The programming solutions are more tedious than are the practical solutions.

Random Scheduling

Another use of PICK was suggested by Brenda Litwin, Logo enthusiast and junior high assistant principal in San Bruno, California. The problem she faced was to assign students to their science, English, and history classes. The students were already assigned to reading and math classes based on placement criterion. But she wanted the composition of the other classes to be as heterogeneous as possible. Her ingenious solution was to create a scheduling program in Logo to generate students' schedules.

The scheduler consists of procedures based on the assigned classes of students. If a student was already assigned to 2nd period math and 3rd period reading, then science, history, and English had to be scheduled for periods 1, 4, and 6. Based on this knowledge, a schedule called SCHEDULE.7C was written.

```
TO PERIOD.ONE
OUTPUT [SCI...JONES HIST...SMITH]
END

TO PERIOD.FOUR
OUTPUT [ENG...KEATS ENG...DUNNE ENG...HARDY]
END
```

Period Six has the same arrangement as Period One, but the schedule would be reversed. If a student gets history in the first period, science would be needed in period six. The PERIOD.SIX procedure takes care of the switch.

```
TO PERIOD.SIX
TEST :ONE = FIRST PERIOD.ONE
IFT [PRINT SE [6] LAST PERIOD.ONE]
IFF [PRINT SE [6] FIRST PERIOD.ONE]
END
```

Now the procedures are used to create a class schedule for someone.

```
TO SCHEDULE.7C :NAME
PRINT SE [SCHEDULE FOR] :NAME
MAKE "ONE PICK PERIOD.ONE
PRINT SE [1] :ONE
PRINT [2 MATH...GAUSS]
PRINT [3 READING...BOOKBINDER ]
PRINT SE [4 ] PICK PERIOD.FOUR
PERIOD.SIX
END
```

This is an excellent example of using Logo to complete a task. The RANDOM operation is a very handy idea with can be used in many ways. Keep looking for new ways to RANDOM. Logo allows you to experiment with this extremely powerful idea.

If you have additional ideas to share, please send them to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or on CompuServe at 76606,1623.

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Conference Update

Logo 86, the third International Logo conference in a series, will be held at the Massachusetts Institute of Technology, July 9 - 11, 1986. Preconference Logo tutorial sessions are scheduled for July 7 - 9. For more information, write to Joyce Tobias, Logo 86, MIT E15-309, 20 Ames Street, Cambridge, MA 02139.

The Educational Computer Consortium of Ohio (ECCO) announces its Spring Logo Conference on Friday, May 9, 1986, in Cleveland, OH. Seymour Papert is scheduled to be the keynote speaker, and a variety of Logo workshops and presentations are scheduled. For more information, write to Alice Fredman, Conference Director, ECCO Center, 1123 SOM Center Road, Cleveland, OH, or call (216) 461-0800.

Keene State College is sponsoring a two-day Logo conference as the culminating activity of a two-week Logo institute. Hosted by Dan and Molly Watt and Tony Stavely, the conference promises to be a lively affair. The program includes presentations by the institute faculty and participants, invited guests, and other Logo-using educators. A schedule of procedure plays, Logo music, and procedural contra dancing with live music will furnish many opportunities for informal learning. The conference is open to all. For more information, write to Tony Stavely, Keene State College, Keene, NH 03431, or call (603) 352-1909.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Jim Fry

When most people think about their musical experiences, they have fond memories of groups gathered together singing or playing instruments. Music has a special ability to bring people together whether in work or play. Jim Fry, a Chapter I math teacher in Plymouth, Michigan, has been experiencing the pleasures of working with other teachers and students to create music with Logo.

Music Hath Charms

Jim's interest in Logo music was stimulated as he began using it to help his remedial math students understand mathematical relationships. He feels that the mathematics inherent in the patterns of music make a rich field for exploration with Logo. Jim and his students have written procedures to make musical phrases and music tools to transpose songs and reverse the order of the notes. In the process, the students and teachers decided to standardize the duration of the notes so that 32 is a whole note, 16 a half note, etc. What better way to help understand fractions?

This year, Jim has been working in a Logo integration project with a group of about 25 other teachers from several school districts in Oakland and Wayne Counties. He has been part of a sub-group working on music. With Austia Hollan and Lary Smith, Jim has completed a two part project, developing music games and music tools.

In the High - Low Game and the Long - Short Game, two students play at a time. One types in the command for notes and gives two numbers. The other student, not looking at the screen, tells which is high or low, long or short. In the Up - and - Down Game, the computer selects a pattern of five random notes, and students decide whether the pattern goes up or down. These games are designed for kindergarten through second grade.

From the Top

The second part of the project has been to create music tools for upper grade students. These tools are procedures but students use them as if they were commands. One procedure allows students to write a tune block. Other procedures transpose, mirror, and reverse the tune blocks created. Students can create a procedure for a whole song by making one or more tune blocks and manipulating the tune blocks with the other tools.

Other groups in the project are creating tools and activities to use Logo to teach skills in math and language arts. Jim's group has designed the procedures to work in Apple, Terrapin and Commodore versions. The Oakland Intermediate School District will soon publish the results of this project with documentation and disks. Ordering information will be published in NLX as soon as it is available.

All Together Now

When Jim first began using Logo, he felt isolated because he didn't know other teachers who were using the language. Jim's solution was to establish Logo Information For Teachers (LIFT), a clearinghouse group. Reggie Bowden and Anne Federlein worked with him to publish a directory of Logo using educators. More recently, the group has organized conferences at the University of Michigan focused on integrating Logo, not just teaching Logo as a programming language. Brian Silverman of LCSI and Tom Lough of NLX were guest speakers this year.

Jim and the LIFT group sponsor the "Teacher to Teacher" column in the NLX, and publish reviews of selected Logo publications.

Jim has also become involved in the CONFER Network set up through the University of Michigan. On the CONFER system, users participate in philosophical discussions as well as sharing practical teaching ideas. Anyone interested in this conferencing network may participate through Telenet or, in Michigan, through the Merit system. Further information may be obtained from:

Professional Development Office
University of Michigan
School of Education
Ann Arbor, Michigan 48103

Cooperation and sharing ideas are inextricably linked with Logo instruction. If you know of someone whose ideas and projects should be shared through this column, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, Texas 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Krell Announces Logo Price Break

Educational institutions and school systems may purchase Krell Logo at $14 per copy in lots of 100 or more. For more information, write to Krell Software Corp 1320 Stony Brook Road, Stony Brook, NY 11790, or call (800) 245-7355. In NY, call (516) 751-5139.

Logo Disserts:
Dissertations Dealing with Logo

by Barbara Elias

A study by John Olive of Emory University may be of interest to high school mathematics teachers. Olive investigated the potential of Logo for developing students' reasoning skills in geometry in a learning environment supportive of relational learning.

Relational learning involves understandings which result from a learner's personal involvement with problem situations and ideas. It occurs when a learner is able to integrate and inter-relate information to solve problems. A synthesis of the *SOLO (Structure of Observed Learning Outcomes)* taxonomy and *Skemp* mathematical learning model provided the frame work for examining how students progressed through a series of instructional sequences in geometry. The taxonomy is a tool for helping teachers organize instruction in a hierarchy of levels of increasing abstractness.

Subjects for this study were 19 ninth grade students, randomly selected from a class of 39 students in a southeastern American community. The group consisted of 13 girls and 6 boys, ranging in ability level from "just below average" to "just above average." Most of the students had little previous experience with computers, and none had prior experience with Logo.

Using a guided discovery approach, students worked 3 days per week for six weeks on a series of Logo projects. They investigated simple regular polygons, variables in procedures, complex polygons, circles and arcs, and recursion. Rotations, reflections, and translations were also explored. A dribble file was made of each student's work. The files of a representative sample of 8 students were analyzed in detail using the *SOLO / Skemp* model.

Independent analyses of days 6 through 12, both by the researcher and by the evaluator taught by him, produced an inter-rated reliability correlation coefficient of 0.93.

Results indicate support of the *SOLO* taxonomy as a useful tool for studying learning processes. Some of the students were able to progress through the instructional sequences and demonstrate understanding of the Logo language and the geometric concepts (such as relational understanding) involved at the higher levels of the model.

Other students, however, showed confusion and regression with the fast pace and progressive complexity of the instruction. They were able to demonstrate only instrumental learning. (Instrumental learning generally involves rote memorization of information acquired from external sources.)

This suggests that instruction that is inappropriately paced may affect how students integrate new learnings. All students appeared to start each learning episode at the lower level and progress through successive levels. They tended to return to lower response levels when tasks became more complex.

Students who demonstrated relational learning, however, subsequently moved through the levels at a faster pace. These students also did more exploring in the early episodes. A number of helpful suggestions is included for workers interested in using dribble file data analysis.

>>>Olive, John (1985). *A Study of the Application of a Qualitative Taxonomic Synthesis to the Analysis of Geometric Reasoning in a COmputer Environment* (Doctoral dissertation), Emory University, Atlanta, GA 1985.

Many thanks to those who contributed dissertations and theses for this column during the year!

(Ed. note: Each month, Barbara Elias highlights a dissertation or thesis dealing with Logo. She is accepting copies of recent research results for the NLX Dissertation and Thesis Repository in preparation for the 1986 - 87 publishing year. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

NL-Textual Contest Winners Announced

Ricky Carter of Lesley College in Cambridge, MA, and Eric Nagler of Santa Clara, CA, have emerged as the winners of the NL-Textual Contest announced by Glen Bull and Paula Cochran in their January 1986 NLX column, "Listful Thinking."

The object of the contest was to create the best procedure or series of procedures which makes use of the fact that a procedure itself is a series of lists.

Both winning entrants wrote programs which print out a master procedure and all associated subprocedures. Originally, there was only one prize announced. Since the two top entries were similar, an additional prize has been awarded.

Ricky Carter is the winner of a Koala Pad.

Eric Nagler wins a copy of Brian Harvey's book, *Computer Science Logo Style*. Eric's program not only prints out the procedures, but also puts a graphics dump on the same page.

Congratulations to Ricky and Eric! Thanks to all those who submitted entries. Thanks also to Glen and Paula for administering the contest. Details on other contests will be announced in future NLX issues.

Listful Thinking

by Glen Bull and Paula Cochran

Turtles and Lists: Never the Twain Shall Meet?

In the call for papers for the Logo 86 Conference, the program chair, Brian Harvey, raised several issues which he hopes will be addressed at the conference. One of these issues was raised by Uri Leron at last year's conference. He suggested that there are two Logos, the Logo of elementary turtle graphics and the Logo of advanced list processing, and that there is no easy way to get from one to the other.

Brian Harvey asks, "What about this? Is it true? Is it okay, or is it a problem? Is there, in fact, a reason to design one language for both ends of the spectrum?"

What's the Answer?

The answer is, "It depends." However, with the right approach, and a version of Logo which has the right features, we think that there is a natural transition from turtle graphics to list processing. In fact, in our introductory educational computing classes, we introduce teachers to lists and Logo poetry in the third week of the class. Turtles and lists become increasingly intertwined from that point to the end of the semester.

Papert talks about using familiar concepts to make new ideas easier to grasp. He calls this the "continuity principle". For example, children know how to move their bodies in space. This knowledge can be translated into a knowledge of how to move the turtle on the computer screen. Thus, Papert feels that turtle geometry provides a better introductory domain for learning mathematical thinking than does arithmetic. (Papert refers to learning something which isn't related to anything you already know as "disassociated learning.")

Turtle graphics is a good introduction to computer programming because it is concrete, and based on concepts which are already understood. It is easy to see what went wrong in a procedure when the turtle turns left instead of right, while the the effects of a command which opens a file are less visible.

Prerequisites to Lists

It is desirable to have turtle graphics and list processing in the same language because it makes it easier to relate events in a list of words to parallel operations performed by the turtle. Thus, once turtle graphics becomes a familiar domain, it can be used as a stepping stone to lists. In other words, turtle graphics becomes a tool which is used to teach the concepts of list processing.

Two of the most important concepts introduced through turtle graphics are:

1. The concept of a procedure
2. The concept of modularity

Both of these concepts are crucial to advanced works with lists. The advantage of encapsulating a series of commands in a procedure is obvious with a shape such as SQUARE. (Well, sometimes it takes just a little coaxing, but everybody eventually gets the idea.) Modularity and the advantages of one procedure calling another become obvious with HOUSE.

```
TO HOUSE
SQUARE
FD 50 RT 90
TRIANGLE
END
```

A third concept which quickly becomes essential is the idea that information can be passed from one procedure to another. (Ed. note: See Sharon Burrowes' article on page 1 of this issue.) The power of passing parameters becomes obvious when one part of a HOUSE changes size (or doesn't, as the case may be).

As soon as Logo learners have had exposure to the concepts of procedures, modularity, and passing parameters, we go on to lists. Lists have gradually been introduced at an earlier and earlier point in the course. Three years ago lists were not introduced until the sixth or seventh week of the semester. Now they are introduced in the third week of the class.

By the third week, the educators in our course have already had some experience with lists. For example, they are already familiar with the PRINT statement:

```
?PRINT [ HI THERE! ]
```

The REPEAT statement also takes a lists of commands as an input.

```
?REPEAT 4 [FD 50 RT 90]
```

More extensive use of lists is introduced with the procedure PICK. As readers of this column know, this procedure randomly picks a word from a list of words.

```
?PRINT PICK [ BREAD MILK EGGS BANANAS]
EGGS
```

It is a short step to creation of a procedure that contains a list of words, as a substitute for typing in the list each time. The advantages of not typing in a long list of words are parallel to the advantages of not typing in FD 50 RT 90 over and over.

```
TO ADJECTIVE
OUTPUT PICK [ RAINY SUNNY WARM BLUSTERY ]
END
```

Now the student can simply type:

```
?PRINT ADJECTIVE
WARM
```

We usually ignore the question of what OUTPUT is doing in the procedure, and so far no one has ever questioned its presence. When Paul Goldenberg publishes his book about Logo and language with its wonderful illustrations of input funnels and output spouts, then we will begin explaining it to beginners. Currently we defer this explanation until later in the semester.

An important point in this process is that we create this list as a procedure rather than as a variable. The reason for this is that, despite Brian Harvey's excellent explanations in *Computer Science Logo Style*, we have difficulty explaining the distinction between dots and quotes to beginners (and advanced users, for that matter). In fact, we find it considerably easier to teach the concept of lists than to teach the concept of variables. (We're familiar with the standard conventions for conveying the concept of a variable.)

As soon as we have ADJECTIVE, or some similar procedure which picks a word from a list of words, we have an excuse to introduce SENTENCE. For example,

```
?PRINT SENTENCE [SPRING IS] ADJECTIVE
SPRING IS BLUSTERY
```

The wonderful thing about introducing the SENTENCE operation this way is that it is resonant with what teachers already know about English sentences. In other words, the command makes sense in this context. Hal Abelson once lamented the use of English words with lists because of all the wonderful things which can be done with lists of numbers. However, once again, we're building a bridge. After a familiar context such as lists of English words is established, the concepts and mechanics of list processing can be extended to the less familiar domain of numbers.

Lists in Context: Language Arts

As soon as SENTENCE is introduced, we move on to Logo poetry, to provide a model of a content-related activity based on list processing. A POEM procedure can be constructed from several sentences, just as a HOUSE procedure can be constructed from several SHAPES. (You will have to create procedures for NOUN and PLACE. Perhaps you will want to expand the ADJECTIVE procedure, too.)

```
TO POEM
PRINT SENTENCE [ SPRING IS ] ADJECTIVE
PRINT SENTENCE [ SPRING IS ] ADJECTIVE
PRINT SENTENCE [ SPRING IS A ] NOUN
PRINT PLACE
END
```

It is important to establish that POEM is a procedure with all the attributes of a graphic procedure such as SQUARE. For example, just as it is possible to put SQUARE in a REPEAT statement,

```
?REPEAT 36 [SQUARE RT 10]
```

it is also possible to put POEM in a REPEAT statement.

```
?REPEAT 4 [POEM]

SPRING IS BLUSTERY
SPRING IS WARM
SPRING IS A BUTTERFLY
UNDER A BUSH

SPRING IS WET
SPRING IS WARM
SPRING IS A TOAD
IN MY POCKET

SPRING IS BRIGHT
SPRING IS FUN
SPRING IS A TULIP
IN THE GRASS

SPRING IS NEW
SPRING IS WARM
SPRING IS A RAINDROP
UNDER A LEAF
```

Lists without Turtles

With adults it is possible to go directly to lists and poetry without the intervening step of using the turtle as a vehicle for introducing the concept of a procedure. We prefer to use turtle graphics first to built a ramp. However, if we are only given two hours to demonstrate the use of Logo in the English curriculum, we might go directly to poetry or conversational dialogs in Logo.

Listful Thinking continued

We have found that, with novices, the minimum prerequisites for writing poetry programs in Logo are:

1. Ability to use the editor to alter a procedure
2. Ability to save work in a file on a disk

The ability to send work to the printer is also helpful. As later versions of Logo have made effective use of all four arrow keys for editing, it has been possible to teach a minimum subset of editing commands in shorter and shorter periods of time. It is much easier to use the arrow keys than to memorize a set of mnemonics associated with control keys. This seems like a small point, but beginners have so many new abstractions presented that this becomes important. We highly recommend more recent versions of Logo which take advantage of all the arrow keys for editing.

These prerequisites assume that novice teachers are given the following:

1. A poetry template or model to follow
2. The procedure PICK
3. Written documentation explaining how Logo poetry can be used in the curriculum
4. Hands-on assistance when help is needed

If these conditions are met, teachers with no Logo background can leave a morning workshop with a usable Logo activity for language arts. We do not recommend morning workshops as an ideal way to introduce Logo (particularly since it is possible to do more harm than good). However, it does demonstrate that, given a familiar domain of activity such as English, lists can be introduced in short order without undue fanfare or confusion.

When we describe POEM and this type of activity to people who are not familiar with Logo, they sometimes say, "Well, I could do that with Pascal (or C or FORTRAN ... or 8080 assembly language)."

We think that it is an instructive activity for them to write an equivalent sort of procedure in another language such as Pascal (we were just joking about 8080 assembly language). The issue is not whether it can be done, but rather the amount of overhead involved in introducing the activity to people who may never have used a computer before, and the ease with which this activity can be modified and extended to additional applications.

Changing Our Tune

Once lists have been introduced to users in the familiar context of English, they can be extended to other domains. We particularly like to use music as a way of extending lists to other areas. In the past, there often was not enough memory left in microcomputers to do anything interesting after music procedures were loaded. Microcomputers now have more memory, and sometimes have special music chips in hardware. This has made it possible to write interesting music procedures which use lists of numbers.

In some versions of Logo, each number stands for a particular note, while on other computers or other versions of Logo, the number stands for a frequency that corresponds to a particular note. For example, the command NOTE [440] might produce the note A above middle C, which has a frequency of 440 hertz (cycles per second). A procedure could be constructed to play a list of notes in this fashion:

```
?PLAY [ 262 262 294 262 349 330 ]
```

This is the first line of "Happy Birthday", and could be put in a procedure:

```
TO LINE1
PLAY [ 262 262 294 262 349 330 ]
END
```

Of course, it is easy enough to write procedures which make use of conventional labels for notes rather than frequencies. However, if the objective is to get the student to use procedures that involve lists of numbers, it is sometimes better to wait until later before making this adaptation.

Music is a natural domain for programming because nearly everyone is familiar with music and notes, and because music tends to have a very modular structure. For example,

```
TO SONG
VERSE1 VERSE2 CHORUS
VERSE3 VERSE4 CHORUS
VERSE1 VERSE2 CHORUS
END
```

To date, we have avoided addressing music in this column because every version and dialect of Logo handles music in a different way. However, music is a very natural bridge to lists and numbers when it is available, and a great deal of experimentation can take place with respect to the numbers used and the pitch heard. A worthwhile project is to ask teachers (or their students) to write a procedure that produces the sound of a siren. How do the numbers of the musical frequencies relate to what we hear with our ears? (Ed. note: See "Teacher Feature" on page 6 for information on a Logo music project.)

From here it is possible to progress to other forms of mathematical activities involving lists. (There is a surprising amount of math in music alone, which may be the reason that, historically, there has been a natural affinity between mathematicians and music.) For example, by returning to turtle graphics, it is possible to

Listful Thinking continued

explore the patterns which can be created by interactions between lists of numbers and the movement of the turtle.

With a few judicious nudges from the teacher, it can be a short step to the use of lists with matrix algebra and other more advanced applications. For a plethora of ideas, we suggest David Thornburg's book, *Beyond Turtle Graphics: Further Explorations of Logo* (Addison - Wesley).

From Turtles to Lists

Is it necessary to have lists and turtle graphics in the same computer language? Obviously we think so. Without lists, the limits of turtle graphics are much more narrowly defined. Without turtle graphics, a concrete bridge to lists is missing. Each benefits from the presence of the other.

Is it true that there is no easy way to get from turtle graphics to lists? We don't believe this is true, if lists are introduced naturally and gradually, side-by-side with turtle graphics. We are quite ready to agree that if working with lists is deferred until all the concepts of turtle graphics are thoroughly explored, there may be trouble when the transition comes.

In teacher training, we introduce lists in the context of a content-related activity, such as POEM. We avoid discussing lists for the sake of lists (as we might in a computer science class). Consequently we do not have to sell teachers on the power of lists because they experience it firsthand.

Why are lists sometimes perceived as difficult? It is easy to forget how short a time Logo has been available on a widespread basis. It takes time to discover natural ways of introducing concepts. Teachers have had much longer to discover the best ways to introduce basic principles in physics and biology. Given a little time and experimentation, effective ways to introduce list processing in Logo will emerge.

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. His CompuServe ID is 72477,1637. Paula Cochran is finishing her doctorate at UVA's Curry School of Education. She is interested in Logo applications in language arts and speech-language pathology.

Logo Tour to Iceland and Holland

The International Logo Exchange announces a Logo tour to Iceland and Holland, August 10 - 23, 1986. Post-graduate credit is available. For more information, write to Educational Spectrum Inc., 188 Summerfield Street, Scarsdale, NY 10583, or call (914) 723-2242.

TurtleTips

by Jane Toth
and Donna Lanyi

Let your imagination soar! By this time of year, our students have become "seasoned Logo veterans," and are always amazing us with new ideas they discover. They especially like time-saving hints, such as the use of the SETPOS and SETHEADING commands.

The Turtle Teacher

As we've discovered, Logo is an excellent means of introducing and reinforcing geometric concepts to our students. They have already been exposed to arcs, circles, lines, and various shapes. Now comes the big challenge - coordinate geometry!

Most mathematics series do not tackle this concept until fifth or sixth grade. With Logo, a coordinate guide, and extra practice, younger students can grasp the concept.

We start off with our "handy dandy" graph paper. Students are shown the coordinates and then asked to number their papers correctly with an x- and y-axis. (Sometimes we laminate these papers for repeated use.) From the beginning, we plot the x-coordinate first, and then the y-coordinate.

Then, students use coordinates to find the starting points for various Logo drawings. For example, instead of using CS PU LT 90 FD 50 RT 90 to begin, they use CS PU SETPOS [-50 0].

On a large screen monitor, we mark points and ask students to estimate the coordinates. SETPOS is used to check the estimates. Once students understand coordinate points, we then demonstrate the usefulness of PRINT POS when trying to create a symmetrical shape.

When introducing the SETHEADING command, we always review the Total Turtle Trip Theorem, using a large laminated circle with degrees marked around the perimeter. Students quickly grasp the idea of turning the turtle to the exact heading rather than guessing the orientation with different intervals of turns. SETHEADING is particularly useful to students who have difficulties estimating turns.

A number of teachers have suggested that perhaps SETHEADING should be taught *before* the RIGHT and LEFT commands are introduced. If any NLX readers have experience with this idea, please share your thoughts with us.

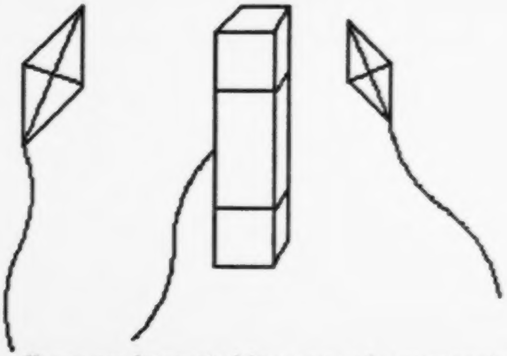
Procedure of the Month

Soar to new heights with Logo! While studying computers in today's technological world, students

TurtleTips continued

need to be reminded that the computer is often used for creating a first draft in the design and engineering trades.

Turn your classroom into a drafting room! What about a Logo kite project? Have students design their own personal kite on the computer. Then help them actually construct their kites. Set a date for "high flying," when the kites are sent aloft from the playground or parking lot. This project contains many related ideas for long-term projects.



If you are interested in a copy of our procedures for kites, please send us a stamped self-addressed envelope.

What interesting things are you doing with Logo as the school year comes to a close? Send your ideas to Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691, or CompuServe 72477,2623.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

NLXionary
A Lectionary of Selected Logo Readings

with Commentary and Opinion
by Griff Wigley

"Logo, A Computer Language at the Crossroads," by David Thornburg, *A+ Magazine*, March 1986.

>David Thornburg is one of Logo's most prolific writers and is certainly a staunch Logo advocate. While he believes that Logo is primarily a language for learning, i. e., a tool for exploring content areas, his love of programming as a skill in and of itself and his knowledge of programming languages in general has always been evident in his writing.

>In this article, Thornburg bemoans the fact that Logo has not become more popular as a programming language. He cites several reasons for its failure to achieve greater respect among programmers:

1. the marketing of Logo as a graphics language,
2. its inexorable ties to Papert's educational philosophy,
3. its reputation for being slow and a memory hog, and
4. it has never been bundled free with computers as BASIC has.

>His hopes are on the rise, however, as Logo interpreters and compilers become available. He explains some of the spectacular results his Stanford graduate students in design were able to achieve using the ExperLogo compiler on the Macintosh, results that they had not been able to achieve using other languages. Some of the students' projects are now in the final stages of commercial development.

>He also contends that people who have become proficient Logo programmers have had a much easier time learning other languages such as LISP, PROLOG, Neon, and Forth.

>Thornburg mentions no ideas for solving this problem, and leaves many questions unanswered. Should schools start buying Macintosh computers instead of Apple IIs? Will the power of the new 16 bit Apple II provide the power needed? Should schools, in the name of programming, undertake the effort to change the emphasis from BASIC to Logo at the secondary level? Would emphasizing Logo's value as a programming language detract from its educational mission?

>If powerful versions of Logo complete with interpreters and compilers become available for the new 16 bit Apple II, then Logo may have a chance with secondary computer science departments. If college and university computer science departments begin using Logo routinely as their "foundation" programming language, and it replaces Pascal as the language for advanced placement, then it could be a shoo-in as the dominant all-purpose programming language.

>But don't bet on it in the near future. Logo's reputation as a language for little kids will not die easily, and that may be the greatest hindrance of all in the esoteric world of computer science.

If you would like to respond to this commentary, or know of other Logo related articles and issues which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

LIFT's
Teacher to Teacher
A Review of New Logo Materials

by James Fry

This month, we look at a general Logo project book, and one with a much narrower focus.

Logo Projects for Everyone

Nudges: IBM Logo Projects, by Steve Tipps, Timothy Riordon, and Glen Bull, 1984, and *Nudges: Apple Logo Projects*, by Glen Bull, Timothy Riordon, and Steve Tipps, 1985, published by Holt, Rinehart and Winston, 383 Madison Avenue, New York, NY 10017.

Target Audience: All levels, from beginner to expert.

Content: The purpose of a "nudge," as the authors state, is to give "a little push in a direction." *Nudges* is a collection of 93 Logo activities, grouped into categories with such inviting names as "Here a Word, There a Number," "Turtle's Secret Codes," "Music Makers," and "What are the Odds?" They follow a common structure in that a little information is given to the student, followed by activities which help the student learn a new concept or skill. Then further explorations are presented and encouraged. Teachers and students are encouraged to take time to explore and discover as much as possible before going to another "nudge." An 84-page teacher's manual is also available, which contains information on the concepts presented, in addition to suggestions for more activities and explorations. The Apple book contains activities for both Apple Logo and Apple Logo II.

Format: *Nudges* is a paperbound book of over 270 pages, and includes a table of contents, an overview of beginning Logo commands, the 93 "nudges," and an appendix on file and workspace management. The IBM book also contains an index for both procedure names and for Logo commands. *Nudges* uses a helpful system of picture symbols (icons) throughout the book for preparation, caution, problems to solve, further explorations, cleaning up the workspace, saving procedures, and other activities.

Strengths: *Nudges* follows an excellent philosophy for working with Logo: give the students a "nudge" to get them started, but do not overstructure the activity so that it is difficult to explore. The effective use of icons is commendable.

Comments: I recommend that all Logo users and teachers of Logo "nudge" themselves to the nearest book store and pick up a copy of *Nudges*. It is an excellent addition to anyone's Logo library.

Logo in the Mathematics Classroom

Mathematics and Logo - A Turtle Trip Through Geometry, by Kathleen Martin and Donna Bearden, 1985, published by Reston Publishing Co., Reston, VA 22090, $19.95.

Target Audience: Middle school students.

Content: The purpose of *Mathematics and Logo* is to help develop mathematics and geometry skills in the middle school. The book contains many activities and problems to allow students to explore and discover concepts involving angles, polygons, circles, and spatial relationships. The activities are described as if you were working with the students. Questions help guide the students to important mathematics ideas. The activities could be used with individuals, small groups, or classroom sized groups.

Format: The book is divided into a short introduction, 10 chapters each covering a different concept, appendices, and a list of resources. Appendix A shows many drawings which could be used with each chapter. Appendix B lists procedures in both LCSI and MIT Logo which could be possible solutions to the suggested problems in each chapter.

Strengths: The book is bound in a plastic spiral so it lies open. It could be used easily in a lab or classroom setting. *Mathematics and Logo* is an excellent resource book which shows how Logo can be integrated into the curriculum. The manner in which the book is written should help make readers feel more comfortable about using Logo. The many questions included should give a number of useful ideas to readers.

Comments: How about more books like this one?! For additional Logo and mathematics ideas, I recommend Martin and Bearden's *Microquests* publication. For more information, write Martin Bearden, Inc., 2502 Peach Tree, McKinney, TX 75069.

What new Logo materials and books do you find useful in teaching? We invite you to send information to the Logo Information For Teachers (LIFT) group, PO Box 5396, Plymouth, MI 48170, or send electronic mail on CompuServe to 76317,565.

James Fry uses Logo with his Chapter 1 remedial mathematics students at Novi Community School, Novi, MI, and is a co-founder of the Logo Information For Teachers (LIFT) group.

Q and A

by Jim McCauley

Q. I heard John Allen speak at a conference recently about something called "first-class objects." I asked around, but no one else seemed to know exactly what these objects are and how they relate to Logo. Can you help?

A. Dr. Allen is well known for his firm and somewhat controversial opinions on Logo, which are brilliantly presented in his book, *Thinking About [TLC] Logo* (Holt, Rinehart & Winston, 1984). If you are serious about Logo, I urge you to obtain and study this book soon as possible.

Design of a Language

Computer languages have many components. Among these elements are variables, constants, literal data, functions, procedures, and data structures. In many languages, each kind of object has different uses and effects. Two important issues in designing computer languages are:

1. How many different types of objects should the language be able to work with, and

2. How should different types (or "classes") of objects be allowed to interact with one another?

A language that adheres to the principles of "first-classness" will have only a few types of objects and will allow them to interact freely with one another.

Languages tend to go to extremes with respect to these issues. In Pascal, for example, there are many different types of objects, and the ways in which they are allowed to interact are complicated and tightly controlled. The designers of Pascal feel that it is very important to keep control of these interactions. Without going deeply into the reasons, this can result in fast, efficient programs that can be carefully checked for errors.

Different Drummer

The Lisp / Logo community has a different orientation. There is only one data structure in Lisp: the list. Nearly all of the objects in a list processor can be used in the same way; that is, every object has a value associated with it, which it can supply when called upon. This allows the very simple syntax of Lisp to do very powerful things. Because any program-part can interact freely with any other program-part (an important element of first-classness), software tools developed in one program can easily be transported to any other program and be expected to work perfectly well.

As for error-checking, the tradition of Lisp puts much of the responsibility for keeping programs bug-free in the hands of the programmer. Also, the power of the Lisp environment actually makes it reasonable to write programs that will check their own consistency and repair themselves when they go astray. Much of the work in artificial intelligence concentrates in this area.

Second Class Logo?

Logo retains much of the flavor of Lisp, although some implementations are closer to the ideal of first-classness than others. Most Logos are divided into commands and operations; this violates first-classness because the two types of objects don't behave in the same way.

Operations produce outputs, while commands do not. When one is doing simple things in Logo, this is not perceived as a shortcoming, but it looms as a more important issue when complicated projects are attempted. For example, if one attempts to add to Logo a new processing function that will accept both commands and operations as inputs, it is usually necessary to write two versions of the new function and add time-consuming code-checking features. If all versions of Logo adhered to the principle of first-classness (for example, by ensuring that both commands and operations produced outputs), this problem would not exist.

It is important to examine issues such as this which relate to the Logo environment. Thanks for your question.

• • • • •

Each month, I will answer a selected question about Logo programming techniques or the thinking which leads to programming. If you have a question, send it to: Jim McCauley, NLX Q and A, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a stamped self-addressed envelope. You may also send questions to me at 70014,1136 on CompuServe's electronic mail system.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

International Logo Exchange Published

If you are an educator who wants an international perspective on Logo activity worldwide, then consider a subscription to *The International Logo Exchange* (ILX) newsletter. This bimonthly publication contains reports from each continent as well as Logo conference and research information. $19.50 per year, $24.50 overseas. ILX, Box 5686, Charlottesville, VA 22905.

NLXual Challenges

by Robs Muir

Round and Round

As I write this, my wife and I are expecting our second child. Our situation has gotten me to thinking about circles lately. Life goes around in much the same way as our Logo turtle. We never go directly to our destination; every step causes us to head off in some new direction; and the final destiny could have been predicted from the start. But the journey is everything.

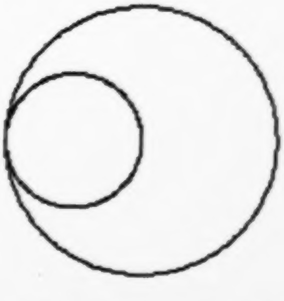
The Circle Primeval

It is interesting and instructive to watch children (and adults) drawing circles in Logo. There is a substantial number of Logo users who seemingly never learn any form of Logo circle other than the one they were first "taught." This usually looks something like

```
REPEAT 360 [ FD 1 RT 1 ]
```

If they need a smaller circle, their solution might be

```
REPEAT 360 [ FD 0.5 RT 1 ]
```



Never mind that the turtle draws the smaller circle in about the same amount of time. For some, the tyrant of Expedience reigns in (and reins in!) the land of Efficiency.

Circular Questions

As a teacher, I always ask two questions of my students as they begin to explore circles and the Total Turtle Trip Theorem:

1. Can you get the turtle to draw the same circles faster?
2. What are the fewest number of sides that you can use and still get an acceptable circle? That is, does REPEAT 15 [FD 10 RT 24] draw as aesthetic a circle as REPEAT 24 [FD 10 RT 15]? (Clearly, this is a function of the resolution of the computer monitor, but the question, "What is the difference, if any, between a circle and a polygon?" cries out to be asked.)

Is there a "Platonic" circle unblemished in its purity, or are all circles simply "hi-res" polygons composed of very short, straight lines? Do the planets and their satellites revolve in circular orbits (ellipses, really), or does differential calculus help us to see the world as it really is - infintesimal fractions of a second during which Newtonian physics demands that objects travel in straight lines?

What is a Circle, Anyhow?

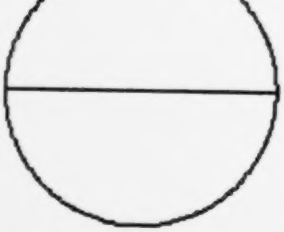
This month's challenge? Decide for yourself whether a circle is fundamentally different from a square. Then help me explore an interesting idea related to Steve Tipps' NLX column last month. (Ed. note: See "Tipps for Teachers: Circular Quirks," NLX March 1986.)

If polygons are merely circles with fewer sides, is it correct to talk about the diameter of, say, a hexagon? Of course, RADIUS, CIRCUMFERENCE, and DIAMETER are mathematical words reserved for circles. But suspend your disbelief for a moment, and let's bend the English language a bit.

Here is a variation on POLY that insures that any POLYgon will have a horizontal line as its bisector, assuming that the turtle begins with HEADING = 0.

```
TO POLY :#SIDES :LENGTH
RT 360 / :#SIDES / 2
REPEAT :#SIDES [FD :LENGTH RT 360 / :#SIDES]
LT 360 / :#SIDES / 2
END
```

Do you see the logic in this POLY? You can draw a "circle" using POLY 360 1 and check its diameter by RT 90 FD (360 / 3.14159).

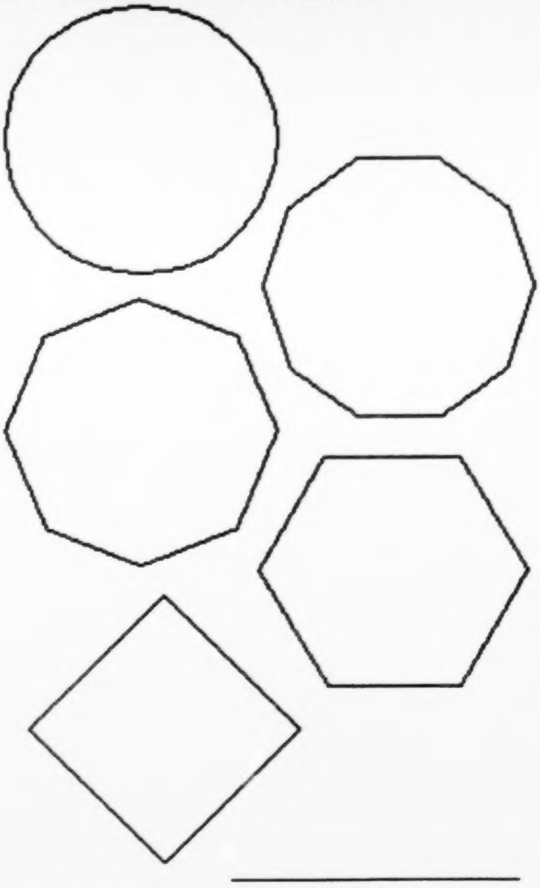


You can also approximate the diameter by simply fooling around with the turtle and carefully measuring the distances involved.

Question: How does this POLY differ from simply typing REPEAT 360 [FD 1 RT 1]?

NLXual Challenges continued

Here is a drawing of a 36-agon, a decagon, an octagon, a hexagon, a square, and a 2-agon (Are we "allowed" to talk about a two-sided polygon?).



In a "pure" circle, the circumference divided by π is said to be its diameter. Build POLYgons with a decreasingly smaller number of sides that all appear to have the same diameter, or in other words, can be inscribed in the 360-agon. Chart these characteristics about your polygons:

Number of Sides	Length of Side	Difference
10	35.5	
		4.5
9	39	
		5
8	44	
...	...	

As the length of the sides continues to increase, the difference between the lengths of the sides increases. Continue this until you have a one-sided(!) polygon. What happens to the differences along the way?

I Wonder...

The wonder of Logo can be found in how it can provoke us to ask questions we might not have asked ourselves before. Good Logo teachers provoke their students to ask the same unanswered questions they ask of themselves. And problem-solving is best learned with problems worth solving.

If you are a serious mathematician with the "definitive answer" to circles vs. polygons, or any other comments, we'd be happy to share your arguments with our readers. For a collection of such discussions, send a self-addressed stamped envelope to:

NLXual Challenges
Attn: Robs Muir
1688 Denver Avenue
Claremont, CA 91711

You may also send other solutions and suggestions for additional challenges to me on CompuServe at 70357,3403.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School. He and his wife, Candi, are celebrating the arrival of a son, Galen Andrew Muir, born February 24, 1986. Welcome, Little Turtle!

The National Logo Exchange, copyright © 1986 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, VA 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223.

Editor..........Tom Lough

THE NATIONAL LOGO EXCHANGE
The Logo Newsletter for Teachers

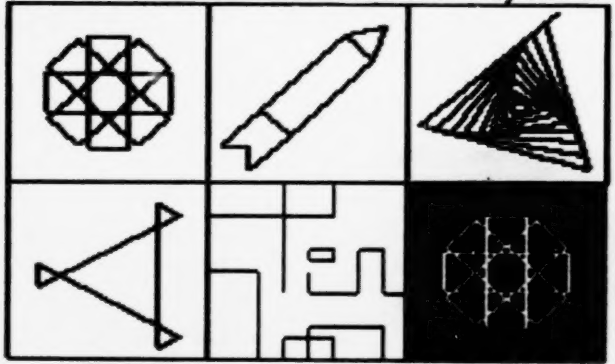
Volume 4 Number 9 FORWARD 100! May 1986 Page 1

Logo Quilt: A Collaborative Learning Project

by
Dan Watt and Molly Watt

The full power of Logo includes:
- the ability to use top-down planning in designing a project,
- the flexible use of procedures, subprocedures, and variables,
- clear naming schemes that make complex procedures easier to read,
- the use of tool procedures,
- the use of modular, state-transparent procedures, and
- an understanding of how to manage the Logo file system and workspace.

Many Logo students do not reach this level of command over the language, however. We have observed that, as long as Logo learners (adults as well as children) work on individual projects, they tend to solve problems in personal, idiosyncratic ways, and avoid learning these powerful techniques. In our experience, people do not learn to use these ideas until they see a clear need for them.



One way to create such a need is to involve several students in creating a collaborative project, in which each one does a part of the programming, working towards a combined result. Ideally, students should have mastered the basics of writing and editing procedures before starting on a collaborative project.

A collaborative project makes an excellent culmination to a semester's activity. If you are looking for a year-end Logo project, here is an idea to consider.

To Quilt Is To Collaborate

One of our favorite collaborative projects is the design of a Logo quilt. We ask each student to contribute an existing turtle graphics design. The group must then create a superprocedure for the entire quilt, and put all the designs together into an overall pattern. The quilt shown here was made by a group of five students. (Ed. note: Full details of this particular project are contained in Chapter 7 of *Teaching with Logo*, by Molly and Dan, Addison-Wesley, 1986.)

Students who put a quilt together must solve a number of problems that lead them to use powerful programming ideas.

First, they must engage in group planning, in order to put the different designs together.

Second, they have to agree on a strategy for naming procedures and subprocedures so that each subprocedure involved in the project has its own unique name and so that all the procedures can be read and understood by all participants.

Quilt continued

One way to accomplish this is to have each programmer add his or her initials to each procedure name. Dan Watt's triangle would be called TRIANGLE.DW; Molly's would be TRIANGLE.MW, for example. This makes it easy to merge workspaces without duplication, but it can fill up the working memory rather quickly.

If similar procedures appear in several individual designs, the group might agree to use one tool procedure called TRIANGLE to eliminate unnecessary duplication. This may be especially important if the group is using a 64K version of Logo!

The Need for Transparency

The quilt project also encourages students to see the value of state-transparent turtle graphics procedures (procedures that begin and end with the turtle in the same position and heading). This is because the simplest way to design a quilt procedure is to have each subdesign begin and end in the same place - for example, in the lower left hand corner of each square.

The use of variables often comes into play as well, as students adjust the sizes of their designs to fit into the squares of the quilt.

Another feature of a quilt design project is that it can be accomplished by students at different levels of expertise. Each student can contribute a design, whether simple or complex. Less advanced students learn more sophisticated programming techniques during the project. More advanced students learn something about the thinking of their less sophisticated partners while appreciating their contributions to the whole project. (This makes quilt projects especially useful for teacher workshops.)

In addition, we observe that "real world" programming projects are more often accomplished by groups of collaborators than by individual programmers. If one of our goals for using Logo is to promote computer "literacy," this gives us another reason to encourage collaborative programming.

Turtle Tradition

We also like the fact that quiltmaking is a traditional art form, with techniques and aesthetics that make use of different materials and technologies. This shows that computer activities do not have to take place in a high-tech vacuum, but can incorporate ideas from other ages and cultures.

Some people go the other way, and transfer Logo quilt designs to cloth for embroidery. At our 1985 Logo Institute, students chose a Logo quilt design as a symbol for the Institute T-shirt, bringing the image full circle. An idea began on fabric, moved to a computer, came off the computer, and back to fabric in a form that symbolized both the beauty of Logo graphics, and the power of group collaboration.

(l. to r., Dan Watt, Molly Watt, Tony Stavely)

Additional Ideas

Other turtle graphics projects are also suitable for collaborative programming projects. Among the ones we've heard about from teachers is having students design different parts of a face (eyes, nose, mouth, etc.) and put them together to make a unified whole. The same can be done with a stick figure of a person, or an outdoor scene involving buildings, trees, flowers, people, animals, and vehicles.

One group of teachers we worked with made an outline of a sweater on the screen, and wrote an interactive "sweater factory" procedure which used the earlier turtle graphics projects done by all the members of the group as designs to decorate the sweater. The same important programming ideas can emerge from any of these projects.

Getting It All Together

Although Logo is known as an "easy" computer language for both students and teachers, we believe that teachers play critical roles in creating learning environments with Logo. Students cannot "discover" everything for themselves. Their teachers must be prepared to coach them through problems, to provide structures, tools, and models, and to know what areas provide the most powerful potential benefits.

Logo has many powerful problem solving ideas built into it. Students might not discover these ideas while working on individual programming problems. Suggesting a collaborative programming challenge is an effective way to help students encounter these important ideas. Why not try it and help your students get the most out of their Logo programming experiences?

Dan Watt and Molly Watt are teachers, lecturers, authors, and consultants in the field of educational computing. Write to them in care of Educational Alternatives, Gregg Lake Road, Antrim, NH 03440.

NLXtra
A Supplement to the National Logo Exchange Newsletter

Volume 4 Number 9 FORWARD 100! May 1986 Page 1

LIFT's

Teacher to Teacher
A Review of New Logo Materials

by
Anne Cairns Federlein

Here are two excellent new Logo resources for a wide age range of children. The following books have been reviewed and tested with elementary school children.

Beginning Students

Logo Works: Lessons in Logo, and *Logo Works Teacher's Manual*, by Sheila Cory and Margie Walker, 1986, published by Terrapin, Inc., 222 Third Street, Cambridge, MA 02142.

Target Audience: Upper elementary children

Content: The authors have researched every component of this book with children and teachers in their school district. In fact, they are to be commended for taking the time to write such a well organized, concise book after teaching all day.

The book is written for use with Terrapin Logo. Each chapter is written in sequential order, with concepts building from skills needed for beginning Logo to problem solving. The seven chapter headings are Getting Started, REPEAT, Writing Procedures, Polygons, Circles, Problem Solving, and Variables & Recursion. The Appendix includes definitions, a command summary, and clear instructions on using this book with Apple Logo, Commodore Logo, and early versions of Terrapin Logo.

An activity disk is included, containing files for Terrapin, Apple, and Commodore Logo.

Strengths: The format begins with introducing upper elementary children to the turtle. Three statements in the beginning give the philosophy of the text and the organization of each chapter.

"Narratives introduce you to something new and important about Logo." Each chapter begins with a narrative explaining the new concepts to be learned.

"Activities give you practice with the new things you have learned." Each chapter gives many ideas for activities in order to provide adequate time for the children to practice the new ideas.

"Explorations inspire you to learn more about Logo and discover on your own." Once the child has practiced the new ideas in the lesson, then open-ended suggestions are given. This allows the child to apply the learning that has taken place.

Older children will find each part of the book stimulating and challenging, especially the section on exploration. Teachers, even the Logo novices, will find the book easy to understand and use as an effective teaching tool.

Intermediate and Advanced Users

Teaching and Learning with Logo, by Allan Martin, published by Teachers College Press, 1234 Amsterdam Avenue, New York, NY 10027.

Target Audience: Middle school and high school teachers

Content: This book contains chapters on the background of Logo, using Logo in the school, turtles and trucks, turtle graphics, making music with Logo, and case studies on thinking about Logo in the classroom. Additional chapters are about the floor turtle, turtle learning in the home, turtles for special children, Logo music in the classroom, and Logo in control. The final chapters are about language with Logo, learning programming with Logo, software in Logo, a case study on Scottish schools one hundred years ago (the author is an educator in Scotland), the future of Logo, and a concluding section on Logo resources.

Strengths: The case studies are wonderful! Allan Martin not only discusses the theory and methods of using Logo in the classroom, but further expands the information by applying classroom practice. All too often in education, the value of the case study is overlooked to the exclusion of theory. The case study method in America seems to be "reserved" for psychology, special education, and medical textbooks.

The chapter on the case study of the Scottish schools of the last century is important because it helps teachers understand the ecology of the classroom and the role of Logo. Subsequently, the chapter on the future addresses the frontiers, new horizons, and the human context.

Martin predicts, "By some future stage, it may well be that Logo is no longer called Logo. Its essential features may be incorporated in a computing environment which will supersede it. Recently developed computer languages like Smalltalk and Boxer enable whole classes of objects with user-defined properties to be created: They could come to offer much in the way of the characteristics of a number of current computer environments. Those

Teacher to Teacher continued

who see the value of what Logo can offer will not mourn the eventual passing of Logo if its successor makes available all that Logo can, plus more, whilst retaining the accessibility and the creative and constructive potential which Logo offers."

This book is highly recommend for all teachers. However, when one reads this text, the question that comes to mind is, "When is the future?" The recent introduction of LogoWriter by Logo Computer Systems Incorporated and Robs Muir's "NLXual Challenges" column on object oriented Logos in this issue seem to suggest that the future is now!

Please contact the Logo Information For Teachers (LIFT) group with information on your favorite new Logo materials and books. Write to LIFT, PO Box 5396, Plymouth, MI 48170.

Anne Cairns Federlein is an associate professor of child development at Oakland University, Rochester, MI, and the editor of *The Journal of the Michigan Association for Education of Young Children.*

NLX Index, Volume 4, September 1985 - May 1986

Articles:

Blitman, Elaine, "Logo Teacher Training," (guest editorial) Nov 85
Burrowes, Sharon, "Procedure Inputs: Local Variables or Not?" Apr 86
Dawson, Sandy, "Online from Paradise: World Logo Conference Report," Dec 85 NLXtra
Harris, Judi, "Euler and the Turtle," Mar 86
Humphreys, Ceil, "Logo Teaching: A Marriage of Theory and Practice," Feb 86
Hurley, James, "Multiple Turtles Without Sprites," Oct 85
Miller, May Lou, "Logo Teacher Training," (guest editorial) Nov 85
Nutting, Kathleen, et al., "Logo 85 Report," Sep 85
Robinson, Louise, "Teachable Logo Moments: Jimmy Comes to Class," Nov 85
Schroeder, Doris, "Come to the LogoShow!" Dec 85
Sharp, Parnelli, and Lovely, Gail, "West Coast Logo Conference Enjoyed by All," Jan 86 NLXtra
Stouffer, Marge, "Sometimes Logo Teachers Should Teach," Jan 86
Tempel, Michael, and Bourgoin, Mario, "IF: At First You Don't Succeed," Sep 1985 NLXtra
Tobias, Joyce, "Logo in the United States: A Survey," Sep 1985
Watt, Dan and Molly, "Logo Quilt: A Collaborative Learning Project," May 86
Watt, Molly, "Twenty Powerful Ideas," Sep 1985
--- "Imagewriter Overprinting Solved for Apple Logo II!" Feb 86 NLXtra

Regular Features:

Bull, Glen, and Cockran, Paula, *Listful Thinking*
- "Anatomy of a Procedure," Jan 86
- "Blending Logo and Language Arts," Sep 1985
- "Instant Tools," Dec 85
- "Logo Word Chemistry," Oct 85
- "Story Time," Feb 86
- "Summing Up," Mar 86
- "Turtles and Lists: Never the Twain Shall Meet?" Apr 86
- "Turtles, Lists, and Spaceships," May 86
- "Words, Names, & Starmaps," Nov85

Elias, Barbara, *Logo Disserts: Dissertations Dealing with Logo*

Fry, James, et al. *Teacher to Teacher: A Review of New Logo Materials*
- Bredehorn, George, *KNOW LOGO: A Logo Exercise Program*, Nov 85
- Bull, Glen, et al., *Nudges: Apple Logo Projects*, Apr 86
- Charischak, Ihor, *Logo Microworld: Picture Puzzles*, Nov 85
- Cory, Sheila and Walker, Margie, *Logo Works: Lessons in Logo*, May 86 NLXtra
- Glantz, John, *Logo Lessons, Volume I*, Mar 86
- Heller, Rachelle, et al., *Logoworlds*, Jan 86
- Hoglund, Barry, *Logo Aide*, Dec 85
- Kwok, Annie, *The Workbook for Learning Logo*, Sep 85
- Marrapodi, Maryann, *Logo for the Apple II/IIe/IIc*, Oct 85
- Martin, Allan, *Teaching and Learning with Logo*, May 86 NLXtra
- Martin, Donald, et al., *Apple Logo Programming Primer*, Dec 85
- Martin, Kathleen, and Bearden, Donna, *Mathematics and Logo - A Turtle Trip Through Geometry*, Apr 86
- Thompson, Patrick, *Seven Microworlds for the Commodore 64*, Mar 86
- Tipps, Steve, et al., *Nudges: IBM Logo Projects*, Apr 86
- Tobias, Joyce, et al., *Beyond Mindstorms: Teaching with IBM Logo*, Sep 85
- Watt, Molly and Dan, *Teaching with Logo*, Feb 86
- Yule, David, *Apple Logo for Kids*, Oct 85

Lough, Tom, *From the Editor*

McCauley, Jim, *Q and A*

Muir, Robs, *NLXual Challenges*
- "The BASIC Challenge," Oct 85
- "Bug Hunting," by E. Paul Goldenberg, Sep 85
- "Ella and the n-gon," Nov 85
- "It's a Plot!" Mar 86
- "It's Only Words," Jan 86
- "Logo Collectibles," Feb 86
- "Making List Operators," Dec 85
- "Objective Orientation," May 86
- "Round and Round," Apr 86

Poplin, Rebecca, *Logo Teacher Feature*
- "Featuring: Peter Anderson," Feb 86
- "Featuring: Jim Fry," Apr 86
- "Featuring: Judith Harris," Jan 86
- "Featuring: Mark Mariotti," Mar 86
- "Featuring: Robs Muir," Sep 85
- "Featuring: Peter Rawitsch," Dec 85
- "Featuring: Sebastian and Sandra Reisch," Nov 85
- "Featuring: Leslie Thyberg," Oct 85
- "Featuring: Sue Wolff," May 86

Tipps, Steve, *Tipps for Teachers*
- "Circular Quirks," Mar 86
- "Community Builders," Nov 1985
- "Dash Away All," Sep 85
- "Geologoboards," Dec 85
- "Getting to Know You," Feb 86
- "Goaltending, Year's Ending," May 86
- "Overland Mapmaking Adventures," Jan 86
- "Random Redux," Apr 86
- "Rooms for Logo," Oct 85

Toth, Jane, and Lanyi, Donna, *TurtleTips*

Wigley, Griff, *NLXionary, A Lectionary of Selected Logo Readings*

From The Editor

by Tom Lough

Turtle Awareness

If all goes well, a turtle will raise her head above the waters near Padre Island in the Gulf of Mexico sometime this month. She will sense a special feeling about the stretch of beach she sees, and swim to shore. Soon, joined by others, she will search for a special location in the sand to dig a bell-shaped chamber, and deposit about 100 leathery eggs.

The special significance about this event is that it will be the first time that a Kemp's ridley turtle *(Lepidochelys kempi)* has ever nested anywhere except a particular small section of Mexican beach. Until recently, the nesting turtles have been at the mercy of poachers and coyotes; the population was driven nearly to extinction.

A few years ago, however, supported in part by a grant from the Exxon Company USA, a project to save the ridley turtle was begun in cooperation with the Mexican government, the National Park Service, the National Marine Fisheries Service, several other federal organizations, and an enthusiastic volunteer organization called Help Endangered Animals Ridley Turtles (HEART, PO Box 681231, Houston, TX 77268-1231).

Each year at nesting time on the Mexican beach, under the protection of armed Mexican troops, ridley turtle eggs were collected as they were laid and swiftly transported to Padre Island. There, the eggs were cared for by interns until they hatched. The baby ridleys were released on the Padre Island sand. Interns stood by as the turtles scurried toward the surf.

After a few moments in the water, the turtles were carefully recaptured and placed in special holding tanks for a year, fed by HEART volunteers. Then, being large enough to have a better than average chance at survival, they were tagged and released for good.

It is hoped that being hatched at Padre Island and crawling across the sand to the Gulf of Mexico waters has imprinted that beach as their future nesting beach.

It is in 1986 that the first significant number of the transplanted ridleys will be sufficiently mature to nest. If the experiment is a success, then the ridley turtle will have a greater chance of survival, with two nesting beaches in operation. We'll know the answer soon.

If you are like me, your level of "turtle awareness" has been heightened by your Logo involvement. You read with interest stories such as that about the ridley. If you see a turtle in the middle of the highway, you stop and rescue it. You collect turtle objects, and now sport a collection of at least a dozen plastic, ceramic, and glass turtles, plus a number made from materials such as walnut shells, and nuts and bolts. (Posy even gave me a topiary turtle!)

This awareness affirms to me once again the richness of the myriad interconnections of the learning process. Interest in one subject can lead to so many others. Time and again, when I observe people using Logo, the learning and experiences lead down unanticipated trails. Exclamations such as, "Hey, this is just like what we learned in ..." are commonplace.

Let's take a lesson from our turtle awareness, and try to be more sensitive to the thousands of additional learning directions Logo can make possible. To be sure, many of them are not in the curriculum. But, like the ridley, each of them is important.

FD 100!

Tom

PS: The mailbag is one of the most important fixtures here at the NLX. This summer, I hope to have time to read. Would you take time to write? Please let me know "how you like your eggs (X)." Is the NLX meeting your needs? What can we do to improve your newsletter?

Tipps for Teachers

by Steve Tipps

Goaltending, Year's Ending

The end of the year can be a time for reflection both for teachers and for students. In the rush to finish chapters and give tests, taking time to ask the really important questions is difficult.

- What have we accomplished?
- What have we learned?
- What can we do now that we could not do in September?

These questions seem essential in the Logo class because it is seen as a place where accomplishment is recorded in terms of personal growth.

Schools, however, generally report accomplishment in percentages right and wrong, grade level equivalences, and standard scores. This conflict between Logo evaluation and school evaluation echoes of the difference between mathematics and school mathematics which Papert discusses in *Mindstorms* (BASIC Books, 1980). This difference is one which Logo enthusiasts must address.

Evaluation Colorado Style

The issue of evaluation served as the basis of a long conversation recently with Forrest Smith of Jefferson County (CO) Schools. Jan Bybee, Kathy Nutting, and Forrest co-ordinate one of the largest Logo projects in the country.

Tipps continued

They want to describe the benefits of Logo they see in the classroom without interferring with the excitement and energy which students and teachers show. They also want to know which classroom practices foster learning with Logo. Putting down on paper what they see every day is the problem.

Goaltending

The first step in evaluation is to be clear on intended goals. Knowing goals is essential in determining what to evaluate. On this point, Logo is a more than a little slippery. It is so flexible and and malleable that educators see different outcomes with Logo. The goals stated for Logo are many:

a. Computer awareness
b. Programming
c. Exploration of ideas in mathematics, English, and other subject areas
d. Problem solving
e. All of the above

The general question of goals with computers in school has often been overlooked in the rush to get equipment yesterday. However, Logo has been subject to scrutiny partially because so many claims have been made for it. From the classroom perspective, the value of Logo can be assessed according to goals of both the teacher and the student.

Computer Awareness Goals

In computer awareness, a teacher might look directly at use of the computer with a checklist.

1. Load a disk
2. Use editing functions
3. Save a program
4. Use keyboard function keys
5. Care of disks and computer
6. Etc.

Obviously, these same items might also be evaluated using word processing, the Koala pad, or other kinds of software. In keeping with the cooperative philosophy of Logo, both the teacher and students keep a checklist and compare their assessments.

Programming Goals

Some people believe that teaching programming is the primary goal of Logo. Many, if not most, teachers begin working with Logo as an "easy language" to teach programming. If programming is the goal, then the evaluation method will involve use of Logo to write programs, to do tasks, and demonstrate skill with the language.

The sophistication of problems and programs can range from "Draw a five-sided figure," to "Write a program which alphabetizes names and averages a list of grades for each person," and beyond.

In the first case, the simple turtle commands are needed with a REPEAT structure. In the latter, lists, functions, recursion, and subprocedures are needed.

Even more specific problems can be written for demonstration of specific programming commands or concepts. For example, "Use input variables in a procedure to calculate interest on a savings account," or, "Write an interactive program which surveys your friends on their favorite foods."

The means of determining whether the programming goal is accomplished is also a checklist.

1. Did the program accomplish the task?
2. Did the program use commands correctly?
    FORWARD, BACK, RIGHT, LEFT
    REPEAT
    etc.

Although many teachers begin with the goal of programming, most soon find that Logo is not so "easy" and that their real goals lay beyond programming for programming sake. The commands were only instrumental to other goals.

One of my goals for students is to alert them to "good habits" in computer use. Some of the good habits are oriented toward programming, but most have wider application and probably belong in the awareness category. Each project includes these particular points for evaluation.

1. Readability - Is the printout organized so that the program can be read and understood? Are the procedure names and variable names meaningful and understandable?

2. Modularity - Does each procedure do a task? Are the procedures and subprocedures organized?

3. Documentation - Are remarks used when needed? Is there a title page in program? Is there a procedure tree drawn? Are pictures provided as needed?

4. Saving - Is the file saved on the class disk? Is the file clean of any extraneous procedures?

5. Operation - Does the program work, or have bugs been isolated?

The checklist is given to students as a way of reinforcing the "big" ideas to consider while working on their Logo projects. Whether students go on with programming or with applications, they are alert to some good habits.

Good habits and language can be evaluated in the short range on a project; however other benefits of Logo programming may also need to be considered. One of the major payoffs for Logo is likely to be in the Logo-to-Pascal link. Although I had Fortran and BASIC, I was a Pascal dropout several years ago.

Tipps continued

This semester, I am working with Pascal again and am finding it so much easier ... almost intuitive. I also see the frustration that students have with concepts of modularity, parameter passing, naming, and structures. They seem to concentrate on the placement of semicolons rather than on structure. These students have more programming experience than I, but their experience does not seem to help them think in Pascal. I believe that Logo has made the difference for me, and that every beginning Pascal course should start with Logo.

Exploration Goals

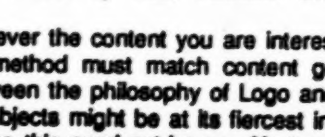
A different goal with Logo is as an exploratory tool in subjects. The natural turtle world holds many fascinating ideas about arithmetic and geometry. Some Logo pioneers have developed tools to explore other content ideas.

Glen Bull and Paula Cochran emphasize language in their "Listful Thinking" column in the NLX; *Microquests* by Kathleen Martin and Donna Bearden (MBI, 2502 Peach Tree Lane, McKinney, TX 75069) is a must for anyone involved in intermediate mathematics. Tom Lough and Jim Hurley have worked with Logo and physics explorations. (See "Logo and Physics" in the January issue of *The Physics Teacher*, and *Logo Physics*, published by Holt, Rinehart and Winston, 1985.)

Whatever the content you are interested in, the evaluation method must match content goals. The conflict between the philosophy of Logo and demands of school subjects might be at its fiercest in this situation; however, this need not be so. No teacher should be comfortable with using Logo, and Logo alone, to explore the rule of 360. Geoboards, string art, protractors, compasses, and the textbooks are also instructional aids for finding out about interior and exterior angles.

Logo is one of many instructional approaches which might be used to explore and learn a concept. But the teacher is the most important element in creating an atmosphere of exploration.

After Logo and many other methods are used and the teacher has had children collect their discoveries, questions on the math test about 360 degrees, size of angles, and the sum of the angles are perfectly appropriate. A question might even pose a Logo-like situation in which an ant was travelling around an irregular quadrilateral and request an estimate of the number of degrees the ant would have to turn at each corner.



The objectives involve degrees and angles and shapes; the evaluation should also. With each content goal, questions can be posed. However, the "fact" questions do not fully assess the benefits of using Logo tools. Part of the result is affective, and that affective dimension should also be addressed with attitude questions. An example appears in the "Logo Teacher Feature" in this NLX issue.

Part of the result is also learning empowerment. A researcher might use the Nowicki-Strickland Locus of Control to assess this (See, for example, Nowicki, S., and Strickland, B. "A Locus of Control Scale for Children." *Journal of Consulting and Clinical Psychology*, 40 (1), 1973, pp. 148 - 154.), but teachers can observe and interview to assess self direction and goal orientation. Such measures may be too "soft" for statistical reports, but they are vital for giving the complete picture of Logo exploration.

Problem Solving Goals

Problem solving behavior is one of the most fascinating constructs in psychological research, and also one of the most difficult to describe, document, and interpret. Is problem solving ability specific to a problem situation or type of problems, or is it generalized as an approach? What are the characteristics of good problem solving, and can they be taught or caught? Although problem solving was one of Logo's greatest selling points, documenting and evaluating this outcome has been difficult.

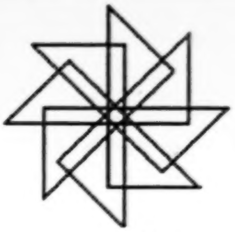
Despite the difficulty, I have ways with which I am content, if not satisfied, for evaluating problem solving. One way deals with planning a task. Logo users seem to look at problems in a more "bite size" fashion.

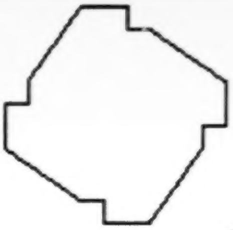
For evaluation of this modular thinking strategy, I have asked students to describe the major parts of a task. Sometimes this is a turtle picture which can be decomposed. Sometimes, the task is an everyday problem, such as how to rearrange the room or complete some class project. Instead of working out all the details first, students look at major tasks. Logo provides a common experience to talk about the strategy adopted.

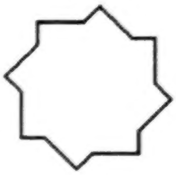
Another problem solving technique is anticipation of results. Can students decompose a problem and see what is going on? A method which is specific to Logo learning is analyzing statements and guessing what they do. REPEAT statements like those in the REPEAT GUESS exercises in *Nudges: Apple Logo Projects* (Holt, Rinehart and Winston, 1985) are good examples of this technique.

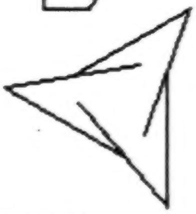
Can you match these statements with the pictures on the next page?

```
        REPEAT 8 [ FD 20 RT 45 BK 40 RT 90]
        REPEAT 8 [ FD 10 RT 45 FD 20 LT 90 ]
        REPEAT 4 [FD 20 RT 90 FD 10 LT 90 FD 10 RT 35 FD
40 RT 55]
        REPEAT 3 [BK 40 RT 140 BK 40 FD 20 LT 20]
```









Evaluation ... and Grading

Evaluation *is* important and difficult, but it is essential in the implementation and improvement of Logo use. Begin with the goals which you have with Logo, and keep evaluation hooked to the goals. Unless you are teaching computer science with Logo, look on Logo as instrumental rather than as the final product. Use evaluation techniques such as anecdotal records, project files, and demonstrations. Checklists and exercises which demonstrate growth in specific areas are a valuable adjunct to demonstrations and files. Include student evaluation as an important element. Avoid grading Logo per se wherever possible. (Thank heavens, Logo does not appear on report cards yet!)

"If I were king for just one day ..." I would do away with grades completely in schools and concentrate on evaluation. Evaluation and grading are related but not the same. Evaluation asks the important questions about learning; grading assigns values to learning, often in an arbitrary manner.

I would like to be clean on the issue of grading Logo. As many of you, however, I have the onerous job of grading courses. I attempt to reconcile my personal dilemma with very specific expectations. I list exactly which commands and which good habits are to be in a project, with a specific number of points for each. However, the projects themselves are completely open-ended, and take on many forms.

The gap between present behavior and belief distresses me. I invite your comments about these ideas. Write to Steve Tipps, Midwestern State University, 3400 Taft Boulevard, Wichita Falls, TX 76308, or send email on CompuServe to 76606,1623.

Have a wonderful summer!

Steve Tipps is the West Professor of Education at Midwestern State University in Wichita Falls, TX, and has been involved with Logo since 1982. He conducts Logo workshops for school systems throughout the United States, and is a popular conference speaker.

Logo Teacher Feature

by Rebecca Poplin

Featuring: Sue Wolff

A few years ago, the sound of pounding hammers and the smell of sawdust signaled that kids were learning in Sue Wolff's Fairfax County (VA) classroom. Her fifth grade students constructed their own special buildings to use in the classroom for different purposes, and learned lots about measurement, planning, cooperation, and hard work in the process. Each building measured about 6 feet by 8 feet, and was made of 2x4's and paneling.

From Dust to Disk

Of course, as soon as computers became available, her students constructed a computer building. Sue realized that the classroom had filled up with buildings and that the carpentry dust wasn't too good for the computers. It was, "Out with the dust, and in with the disks," as the new technology began to replace the old.

Five years ago, Sue began borrowing computers to use in her class. Having started as many of us did with BASIC, she moved first to PILOT, and then to Logo. Fascinated with the types of thinking she observed, Sue wanted to provide her students with a more intense involvement with Logo. Her first step was to write a proposal called Learning Through Logo (LTL). She requested administrative backing for a curriculum which would have computers as the centerpiece of a creative interdisciplinary learning program.

LTL School Room

Before the 1983-84 school year began, Sue organized a meeting for parents of rising fifth graders. Sue explained her project and invited students to participate. That first year, another fifth grade was used as a control group. Sue spent two weeks at the beginning of the year teaching Logo. In the mornings, students participated in content area discussions, research in the library, and teacher-directed instruction, but very little traditional seat work.

Whenever students showed they had studied a topic thoroughly and had lots of information, they would write computer programs in the afternoon. Once or twice a week, students had additional lessons in Logo.

In order to study a topic, students had to submit a proposal. For example, a group or individual might choose a topic like the respiratory system, pick out the most important information, put the information in the computer, and illustrate the report using turtle graphics. Occasionally, the topic had to be narrowed down, as

Feature continued

when a group of students proposed studying "history." Students also wrote math and reading lessons, and made computerized mad libs related to social studies and parts of speech. In the course of the year, Sue's class produced quite a bit of software.

Many of their programming efforts and experiences were documented on videotape which was used for presentations to parents. Sue also showed the videotape as part of her presentation at Logo 85.

Doing Logo Makes a Difference

During the first year of the LTL project, Sue noticed many effects of the intense technology-based study. Some results were nebulous, others more tangible. An immediate outcome was that students produced long stories for their parents when asked, "What did you do in school today, dear?"

An attitude survey showed differences between the LTL students and the control class. Among other effects that didn't show up on tests and surveys, the school librarian began to notice differences in self-direction in Sue's students. Even though the curriculum had not followed the strictest topic guidelines, students in the program experienced more growth in standardized test scores than the other class. During the second year, the program expanded to include the other fifth grade.

By this time, all of Forestville Elementary had gotten involved with computing. As student government fund raisers, students sold purple T-shirts proclaiming, "I'm a computer kid." (Sue is a T-shirt enthusiast. Whenever visitors were scheduled to observe the LTL program, she and all the students wore their special shirts with turtles on them!)

Additional Challenges

Sue has worked as a teacher trainer in Fairfax County and has taught several Logo courses. All teachers have at least an hour a week using the school lab. Some do word processing, others use software, but most choose Logo at least part of the time.

Recently, Sue has taken on a new challenge. She has become a computer resource teacher, working with approximately 45 area schools at all levels. Many of the teachers she helped get started in Logo will get the opportunity to work with her more often.

Sue wants to challenge teachers to use more text with graphics in Logo. She feels that text can help to sell a Logo program, because students can improve their writing and because text helps Logo fit in with the curriculum. With text, students and teachers have a hard copy of something that looks very much like "regular school work." Sue Wolff believes that, when the computers are in place and programs are going, *then* curriculum revision can take place.

If you know a teacher who should be featured in this column, please contact:

Rebecca Poplin
2421 Fain Street
Wichita Falls, Texas 76308

Rebecca Poplin uses Logo to teach computing and mathematics at a junior high school in Wichita Falls, TX.

Logo Notes

Academia de la Investigacion Cientifica announces the III International Symposium on Computers and Children's Education, to be held October 25 - 31, 1986, in Puebla, Mexico. At last year's symposium, a number of the presentations were on Logo. Here is an opportunity to learn about Logo use from a different cultural perspective without the expense of overseas travel. For symposium information, write to Academia de la Investigacion Cientifica, Insurgentes Sur #949-9, Mexico D. F. 03810. Copies of the proceedings of last year's symposium are available for 3000 pesos from Dr. Jorge Bustamente at the above address. Daniel Orey is maintaining a mailing list of Logo users from the USA and Canada who are interested in attending, and will be working on a group discount for travel and accommodations. Write to:

Daniel Orey
CIIMTE - COE
The University of New Mexico
Albuquerque, NM 87131

MACUL SIGLOGO Formed

During the final session of the recent Michigan Association for Computer Users in Learning (MACUL) conference, a group of thirty-five Logo enthusiasts assembled for the purpose of establishing a Logo Special Interest Group (SIG). Christine Canning chaired the informal meeting with assistance from Reggie Bowden, Karen Bunting, Anne Cairns Federlein, Jim Fry, Lynn Lieberman, and Terry Rudman.

After some introductory discussion, the group voted to take the steps to establish itself as an official MACUL SIG. Tom Lough, editor of *The National Logo Exchange* newsletter, spoke briefly, commending the group for its spontaneous generation and enthusiasm. He encouraged the SIG to disseminate its news and information, as well as to provide a model for other groups across the country.

The SIG then divided into three discussion clusters, to explore SIG objectives, activities, and conference planning.

For more information and membership applications, write to Chris Canning, MACUL SIGLOGO, PO Box 628, Westland, MI 48185.

Listful Thinking

by Glen Bull and Paula Cochran

Turtles, Lists, and Spaceships

In our previous column, we considered a question that Brian Harvey raised for participants attending the Logo 86 conference to be held this summer: "Is it important for the same language to contain both turtles and lists?" We said that we thought it was, and added that very often the turtle can be a bridge which makes the effects of list operations more tangible.

In this column, we would like to look at an example of how the two capabilities can complement one another. Generally, we introduce lists through examples which use words and sentences, since everyone constantly uses sentences in everyday life. Once words are used to introduce the concept of lists, through poetry and conversational dialogs, examples involving lists of numbers can be incorporated.

Computers have the potential for making subjects interesting that would otherwise be boring. You would expect that a math major would be interested in trigonometry. The trick is to make the same subject interesting to someone whose first love is art or history. Papert claims we all have the inherent ability to become interested in subjects such as trigonometry, if they are introduced in an appealing way.

In a recent seminar, we asked all who were interested in trigonometry to raise their hands. Only two people did so (both were math teachers). Everyone else in the seminar had a reaction that ranged from disinterest to downright aversion. We have found this is not unusual.

Math and Original SIN

Yet, trigonometric functions, such as the sine function, can be fascinating if they are explored with the aid of a computer, rather than through the static pages of a book. Even if trigonometry is not your first love, please don't stop reading. Most of the procedures are only three or four lines long, and they are almost identical to the ones we used for Logo poetry in earlier columns. (What we're trying to say is that we have a bridge we'd like to sell you ...)

The SIN operation is the Logo equivalent to the sine function. Every angle has a SIN associated with it. (We know your mother told you to avoid sin. However, this SIN is pronounced "sign," not "sin.") There is a pattern to these sines. This procedure will print the sines associated with the degrees in a circle. (In all of the examples which follow, omit the brackets around STOP if you are using Terrapin or Commodore Logo.)

```
TO SHOW.SIN :DEGREE
IF :DEGREE > 360 [ STOP ]
PRINT SIN :DEGREE
SHOW.SIN :DEGREE + 20
END
```

You can run the procedure by typing this:

```
?SHOW.SIN 0
0.0
0.342020
0.642788
...
```

The SHOW.SIN procedure prints a series of numbers that look rather forbidding, primarily because they are long decimals. It's easier to see the pattern if we change the procedure so the numbers are expressed in turtle steps. We can do this by multiplying each sine by 100.

```
TO MAGNIFY.SIN :DEGREE
IF :DEGREE > 360 [ STOP ]
PRINT ROUND 100 * SIN :DEGREE
MAGNIFY.SIN :DEGREE + 20
END
```

Now the numbers produced are more like ones you might use with the turtle.

```
?MAGNIFY.SIN 0
0
34
64
87
98
...
```

Turtle on a Pogo Stick

In fact, imagine that the turtle was moving up and down from one point to the next. It would start out at 0, move up 34 turtle steps, then go to 64, and so on. Here is a procedure to move the turtle in just that way. We've multiplied each sine by 50 instead of 100 to keep the turtle from going off the bottom of the screen in the splitscreen mode.

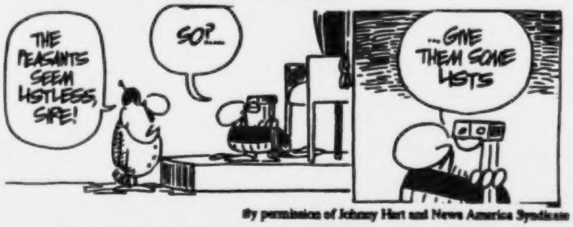
```
TO POGO :DEGREE
IF :DEGREE > 359 [ STOP ]
SETY 50 * SIN :DEGREE
POGO :DEGREE + 10
END
```

We called the procedure POGO because it makes the turtle bounce up and down almost as though it were on a pogo stick. In fact, the turtle behaves very much as though it were on a spring. It slows down at the end of the motion and speeds up as it moves through the middle. Is this just an optical illusion? Try it, and see what you think.

```
?REPEAT 3 [ POGO 0 ]
```

The Turtle Seems Listless

Now that you've got an idea of what a SIN function can make the turtle do, let's add new capabilities by drawing upon the power of lists. Thus far we've been asking the computer to calculate each sine over and over again. The time required to make each calculation slows the turtle down a little. In fact, its movement is halting and, well ... just a little listless.


THE PEASANTS SEEM LISTLESS, SIRE!
SO?...
...GIVE THEM SOME LISTS

By permission of Johnny Hart and News America Syndicate

We can make the pogo stick bounce faster if we calculate the sines just one time, and store them in a list. Of course, you could write down the list of all the sines used and then type them back in with a MAKE statement. However, it's easier to let Logo do the typing for you. Here is a procedure to store the sines used by the turtle in a list:

```
TO CREATE.SINES
MAKE "SINE.LIST [ ]
DO.SINE.LIST 0
END

TO DO.SINE.LIST :DEGREE
IF :DEGREE > 359 [STOP]
MAKE "SINE.LIST SE :SINE.LIST SIN :DEGREE
DO.SINE.LIST :DEGREE + 10
END
```

Now type

```
?CREATE.SINES
```

Logo will go away for about 30 seconds while it makes the list. Although this seems like a long delay, it's probably faster than you could type the numbers in.

Inspecting the List

When Logo finishes the CREATE.SINES procedure, you can see if it worked properly by printing the contents of the list.

```
?PRINT :SINE.LIST
0 0.173 0.342 ...
```

The numbers are printed across the page in a rather messy fashion. A number of columns ago, we demonstrated the use of the INSPECT procedure to inspect Christmas lists, grocery lists, etc.

```
TO INSPECT :LIST
IF :LIST = [ ] [ STOP ]
PRINT FIRST :LIST
INSPECT BUTFIRST :LIST
END
```

We can use the INSPECT procedure to make a more careful inspection of the list of SIN numbers.

```
?INSPECT :SINE.LIST
0
0.173
0.342
...
```

We also wanted to use the INSPECT procedure here because one small modification will allow the turtle to use the list of SIN numbers to create the same pogo stick effect presented earlier. However, now the turtle should move faster and more smoothly since there are no longer any delays while the sines are calculated.

```
TO BOUNCE :LIST
IF :LIST = [ ] [ STOP]
SETY 50 * FIRST :LIST
BOUNCE BUTFIRST :LIST
END
```

At this point we are using the familiar INSPECT procedure to perform an operation on a list of numbers instead of words or sentences. The only difference is that we are replacing the PRINT command with the SETY command to move the turtle. See if the turtle moves more smoothly when a list of numbers is used. Its performance should no longer be as listless when you use the BOUNCE procedure.

```
?REPEAT 3 [ BOUNCE  :SINE.LIST ]
```

Springtime

It is clear that the springiness of the turtle is related to the SIN function. Consider other ways that the turtle could move. It could move forward at a constant speed and then stop instantaneously. Here is a procedure that makes the turtle do that:

Listful Thinking continued

```
TO FLIP
FD 50 BK 50
END
```

Compare these two ways of moving the turtle.

```
?REPEAT 3 [ FLIP ]
?REPEAT 3 [ BOUNCE :SINE.LIST ]
```

(This is a visual concept. You'll have to run the procedures to see what we mean.)

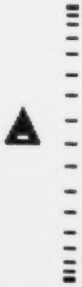
The Plot Thickens

The INSPECT procedure prints the numbers, but the turtle can be used to plot the numbers. The PLOT procedure is just like BOUNCE, except that it leaves a dash at each point along the turtle's path.

```
TO PLOT :LIST
IF :LIST = [ ] [ STOP ]
DASH
SETY 50 * FIRST :LIST
PLOT BUTFIRST :LIST
END

TO DASH
PD SETX XCOR + 2
PU SETX XCOR - 2
END
```

Now type

```
?PLOT :SINE.LIST
```



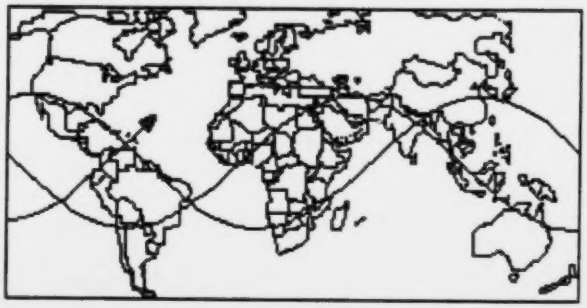
The dashes left by the turtle immediately make it apparent that the turtle really is moving a greater distance between points as it goes through the middle. One of the seminar participants had suggested that the faster movement of the turtle through the middle was an optical illusion. However, the dashes made it clear that the turtle really was moving faster in the middle, and slowing down near the top and bottom.

This illustrates how the dynamic movement of the computer can make concepts clear that are difficult to grasp on a static printed page. (In fact, if you didn't quite follow our explanation, we encourage you to go to the computer and run these procedures.) In addition, the concept is easier to grasp because the computer performs the mechanical drudgery of calculating the points. Even if you were to perform the same task with pencil and paper, the tedium of doing the calculations by hand would take the fun out of it.

Another concept of learning emphasized in *Mindstorms* is the idea of connectedness. It's easier to understand an idea if you can relate it to something you already know. One student confessed that she had plotted (did she mean plodded?) through hundreds of SIN calculations without ever understanding that there was a relationship between this function and springiness. (The actual term is "simple harmonic motion", but rather than let these words get in the way of our understanding, we will simply say that we are talking about the kind of movement represented by a spring or a swingset or a pendulum. For example, think of how a swing slows down at the top of its arc, and speeds up as it goes back through the middle.)

Turtles and Spaceships

This same kind of motion is also associated with many kinds of circular movement ranging from the movement of a bicycle wheel to the orbit of a spacecraft. Possibly you can visualize the path that a spacecraft takes as it moves around the earth ... perhaps you have seen this on a telecast of a NASA mission. Very often, the path of the spacecraft is shown as it moves across the United States, Australia, and other countries.



With a slight adjustment, the turtle can plot a similar path. The only difference between the PLOT and the PATH procedures is that we will move the turtle over about four steps after each point is plotted.

```
TO PATH :LIST
IF :LIST = [ ] [ STOP ]
DASH
SETY 50 * FIRST :LIST
SETX XCOR + 4
PATH BUTFIRST :LIST
END
```

The PATH procedure shows the S-like path that a spacecraft will take as it moves across the world map.

```
?PU SETX -139
?REPEAT 2 [ PATH :SINE.LIST ]
```

To see what happens with successive orbits, clear the screen and try a larger number of repetitions.

```
?PU SETX -139
?REPEAT 5 [ PATH :SINE.LIST ]
```

Listful Thinking continued

If you are interested in pursuing the idea of turtles and orbits, "Kepler" by Jim McCauley is a classic article on this subject. (*The Computing Teacher*, December / January 1983-84, p. 15-22.)

Summary

There are several points that we would like to make. The first is that the transition to use of lists can be fairly natural and painless. The second is that turtles and lists are natural partners. The third is that computers can be used to explore a variety of subjects in ways that are difficult to pursue with pencil and paper alone. The best way to explore a dynamic concept is with a dynamic medium.

We are concerned about the ways that computers and Logo are being used in schools. Teachers very often are being required to teach Logo to fulfill "computer literacy" requirements rather than using the computer and Logo to do a better job of teaching traditional subjects.

Paul Goldenberg has introduced the idea of the "least intrusive" addition of the computer to the classroom. This means that the computer should be used to allow a good teacher to teach more effectively -- but without disrupting her previous teaching style. ("Learning to Think Algebraically: Word Problems and Data Flow." E. Paul Goldenberg, *Proceedings of the World Logo Conference*. Paradise Valley, B.C., October 25-27, 1985.) That is, the computer simply becomes another tool (although a very effective one) rather than a new subject to be learned.

We have expressed this idea as the concept of "using Logo to teach, rather than teaching Logo." The title of this column has been "Listful Thinking," but it could as easily have been "Using Logo to Teach without Years of Study or a Degree in Quantum Mechanics." Lists are just one of many elements in Logo which make it possible to do this.

We wish you a listful summer!

Glen Bull is a professor in the University of Virginia's Curry School of Education, and teaches Logo courses at both the graduate and undergraduate level. His CompuServe ID is 72477,1637. Paula Cochran recently completed the requirements for her doctorate at UVA's Curry School of Education. She is interested in Logo applications in language arts and speech-language pathology.

LogoWriter Workshops and Classes

If you are planning a course or workshop on LogoWriter, and desire assistance with publicity, course outlines, and other support materials, write to Michael Tempel, LCSI, 555 West 57th Street, Suite 1236, New York, NY 10019, or call (212) 765-4780.

TurtleTips

by Jane Toth
and Donna Lanyi

Since our drawing with Logo is just "sailing" right along, it is time to add another challenge. Those who have requested our monthly procedures realize that we usually included a printed message below the picture. Besides adding words to graphics, the use of the PRINT command opens new avenues for our students.

The Turtle Teacher

We always enjoy adding a "cute-sie" phrase to each of our monthly procedures. We do this by using the SETCURSOR command along with a PRINT statement. Here are some ideas for using this technique.

Involve your class in a group comic or picture book. Each student or group could be responsible for completing a picture and a PRINT line to accompany it. Students could write letters, poetry, vocabulary sentences, etc., using the PRINT statement, and illustrate them with a Logo graphic.

We decided to pick topics that we were studying in class from various subject areas - horses from reading, human body systems from health, and US states from social studies. This is definitely a long range project which enables students to work towards a given date.

A useful strategy for working with the Logo text editor and the PRINT statement is detailed in Glen Bull and Paula Cochran's "Listful Thinking" column in the February 1986 NLX.

Also, the exciting new LogoWriter from Logo Computer Systems Inc. (LCSI) would be a natural choice for projects such as this.

(We have heard about people using the Logo editor as a word processor. We have never tried this, but, if a commercial word processing disk is not available in your school, this might be an option to explore.)

From the Mail Box

Phyllis Richardsworks as a computer teacher in the Elmont (NY) Schools. Phyllis sent us a suggestion she found most helpful after reading and using information in Glen and P's excellent "Listful Thinking" column in the December 1985 NLX.

In using their "Instant Tools," Phyllis encountered a problem with the procedure used to quit the program. The children were to press the Q key whenever they wanted to stop. She found that her children lacked keyboarding experience, were easily distracted, and would often press the Q key by accident. Phyllis changed the quit command to a shifted character, requiring the deliberate use of the shift key.

TurtleTips continued

```
TO $
PRINT [ QUIT INSTANT ]
PRINT [ TYPE DRAW FOR REGULAR LOGO ]
TOPLEVEL
END
```

This procedure has eliminated any accidental quitting or other disappointments. Phyllis also included the use of the PRINTPICT command in her Instant, so that she can help her students easily produce printouts of their work. PRINTPICT is in a file on the Utilities II disk for Terrapin Logo, and requires an Imagewriter. She placed the command inside another procedure which uses the shift key.

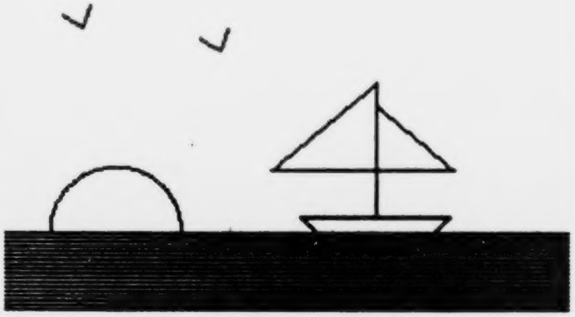
```
TO !
PRINTPICT 1 0
END
```

Thank you, Phyllis, for sharing these ideas!

Procedure of the Month

During May, as students' minds drift toward summer, use their creative thoughts to design a spring or summer bulletin board. Have students design their own sailboats or hot air balloons. Add some poetry or creative writing stories and you have a great eye-catching bulletin board.

If you are interested in a copy of our procedures, please send us a stamped self-addressed envelope.



We have enjoyed working with you this school year. Special thanks to all those who wrote to us. Have a great summer, everyone! Jane Toth and Donna Lanyi, Beall Avenue School, 716 Beall Avenue, Wooster, OH 44691.

Jane Toth and Donna Lanyi use Logo in their elementary classrooms in the Wooster (OH) City Schools.

NLXionary

A Lectionary of Selected Logo Readings

with Commentary and Opinion

by Griff Wigley

"An Interview with Seymour Papert," by Fran Reinhold, *Electronic Learning*, April, 1986.

>In case you hadn't heard, Seymour Papert has a new Logo. And when Papert speaks, the educational computing magazines listen. Although both of the following articles on LogoWriter are authored by Papert, the philosophical underpinnings he discusses are substantially different.

"The Next Step: LogoWriter" by Seymour Papert, *Classroom Computer News*, April, 1986.

>Papert discusses a "first tier use" of a technology, i.e., people using technology to do better what has been done all along. The use of computers for the drilling of basal reading skills is an example of a first tier approach; it improves the delivery of an otherwise unchanged curriculum. Word processing, surprisingly, is also a first tier use; it is nothing other than a high powered typewriter (a use he emphatically supports).

>Logo's turtle graphics is a second tier use of computers because it presents a distinctly different approach to math from the traditional curriculum. Papert says that LogoWriter is a second tier successor to writing with a word processor. It allows students to manipulate text in a natural fashion, just as Logo allowed the manipulation of graphics. It is a programmable word processor.

>By allowing students to actually change and / or create new word processing commands, Papert remains true to his Logo philosophy of putting the student in control over the computer. (This feature sounds like a "macro" capability, common to many of the more high-powered application packages.)

>LogoWriter has the ability to combine text and animated graphics, as well as to control the position and "tempo" of the text. This offers the student a medium to work with that's between text and speech, Papert says. It's a tool for any wide-eyed story-telling child who needs help putting the same feeling into his or her writing.

>Papert hopes that teachers who are more drawn to verbal or dramatic content rather than mathematical subhect areas will now have access to the Logo culture. And as for the "transfer" question, he's convinced that "your imagination as a teacher will show you how to use LogoWriter programming as a transition to pure writing."

NLXionary continued

"New Views on Logo," by Seymour Papert, *Electronic Learning*, April, 1986.

>Papert's approach towards explaining LogoWriter in this article is to compare the learning styles of two children.

>Jessie is a planner, attracted to precision and detail; Jessie could become deeply involved with the original Logo in part because its precision-minded turtle lends itself to planning.

>Pat is your stereotypical artist, drawn to color, motion and a plan-as-you-go style of working. Logo in most of its forms is more limiting to a student with this style.

>LogoWriter is Papert's solution to providing a richer environment for the Pats of the world, while still accomodating the Jessies. It continues to provide a microworld of graphics while adding a microworld of text to explore.

>Logo's extensibility allows the creation of new commands by writing procedures for the manipulation of turtle graphics. LogoWriter's extensibility allows the creation of new word processing commands for the manipulation of text. Papert maintains that a word processor should not be an unchangeable "black box" because then *it* is the master, not the user.

>LogoWriter sounds like an improvement over other attempts to combine text and graphics (Kidwriter, Story Machine, Story Maker, Bank Street Story Book, etc.). Its text programmability feature seems the most exciting. Look for reviews of this software in virtually every educational computing publication, including this one.

>One final note: what assumptions might you make about yourself as to your presumptions of the sex of Jessie and Pat? You will find some excellent food for thought on this matter in Sherry Turkle's *The Second Self* (Simon and Schuster, 1984).

If you would like to respond to this commentary, or know of other Logo related articles and issues which may be of interest to our NLX readers, please write to: Griff Wigley, 918 College Street, Northfield, MN 55057, or send electronic mail to me on CompuServe at 72007,24.

Griff Wigley is a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

Logo Teachers Wanted!

Interested in teaching Logo at a camp this summer? If so, then send your resume with references to Computer Summers, Inc., PO Box 436, South Salem, NY 10590. Camps are located throughout the USA, so specify your geographical preferences..

Logo Disserts:
Dissertations Dealing with Logo

by Barbara Elias

A thesis by Linda E. Morecroft, titled *A Relative Motion Microworld*, presents an example of a Logo microworld designed to teach concepts in relative motion to students in mathematics and physics. Its purposes are to enhance student understandings of relative motion concepts and to provide a rich medium for representing motion in a variety of settings.

Relative motion problems deal with how moving objects appear in varied reference frames and from varied perspectives. For example, the path made by a coin which falls from the hand of a person running across a yard and the way in which pilots determine where to drop relief packages to specified targets on the ground involve relative motion.

Traditional approaches to teaching this topic generally utilize static representations. Such representations are limited for providing students with the kinds of tangible experiences available through computer simulation.

Four basic steps were used to set up the motion simulation:

a. Independent objects were created with the capability of moving on the screen
b. Each object was then told how to move
c. One object was chosen as the reference frame object
d. All objects were set in motion

Problems and explorations in rectilinear and non-parallel rectilinear motion, circular motion, and motion under gravity were prepared and presented in a student text. A teacher's guide was also prepared. The program was developed with Terrapin Logo and used on a 64K Apple II computer. It was also developed in LCSI Logo and used on the Macintosh 512K computer.

The microworld was tested with two tutorial groups of 2 and 3 students each and with a class of 22 students, most of whom were 12th grade physics students. A screening test was given as a pre- and post-assessment. Personal observation, discussions with students, and information from a student survey provided data for the study.

Students spent about one and one half weeks using the materials during their regular school day. The researcher assisted students as necessary. Valuable feedback helped to refine and improve the materials. Most students found the materials helpful, and agreed that their understanding of relative motion had been improved. Post-test scores indicated improvement and students enjoyed using the computer in physics class.

Logo Disserts continued

Physics and mathematics teachers may find the Student Text and the Teacher's Reference Manual especially useful for teaching the topics. In addition, a number of suggestions have been included for teachers interested in developing their own computer based learning environments.

>>>Morecroft, Linda E. (1985). *A Relative Motion Microworld* (unpublished masters thesis), Massachusetts Institute of Technology, 1985.

(Ed. note: Barbara Elias is accepting copies of recent research results for the NLX Dissertation and Thesis Repository in preparation for the 1986 - 87 publishing year. Reports may be mailed to her in care of NLX, PO Box 5341, Charlottesville, VA 22905.)

Barbara Elias is an assistant professor in the Education Department of Virginia State University in Petersburg, VA, and a doctoral candidate at the University of VA.

Q and A

by Jim McCauley

Q. I heard someone say the other day that Logo doesn't have a "null." What does that mean? Does Logo need one?

A. "Null" is one of those words that has many meanings in the world of computing. For example, there is an ASCII character called "NUL" which is ASCII character 0. Ordinarily, it is a character that cannot be printed, since it has no defined printed character associated with it. It is used as a control code. Your version of Logo may print out some character for it anyway; try typing PRINT CHAR 0 and see what you get.

Plenty of Nothing

The usual meaning of the word "null" is "empty; containing nothing." In most versions of Pascal, *null* means two things: a *null statement* (that is, a statement that means and does nothing) or a *null string* (a string containing no characters).

It may seem silly to have statements and strings that are empty, but they are very useful. For example, if you are writing a data-entry routine, it's handy to define the act of typing nothing and simply pressing the RETURN key as a "null string," which could be used to indicate that you are finished entering data.

Logo has two predefined "empty objects." The empty list ([ ]) is enormously useful as a end-test for recursion. It can be used as a means of detecting if a procedure has "BUTFIRSTed" a list down to the point where it contains nothing. The empty word (") can be used for similar purposes in procedures that are recursively decomposing words.

In many versions of Logo, if you type SHOW READLIST and press RETURN twice, you will see that the empty list is defined as Logo's standard "null object". (MIT Logo users, you will get similar results if you type REQUEST and press RETURN twice; you should get RESULT: [ ] as a response.)

NULL and NIL

The use of the empty list as a null object is part of Logo's heritage from Lisp. There is a common Lisp function called NULL that is ordinarily used to identify empty lists (like Logo's EMPTYP or EMPTY? operation).

The NULL function illustrates an interesting difference between Lisp and most versions of Logo. Logo uses two special words (TRUE and FALSE) to identify Boolean conditions. Lisp has two similar words (T and NIL), but these Lisp words have special extensions to their meanings. In Lisp, any empty list is defined to be NIL (false), while anything that is not empty (a word or a list containing one or more objects) is defined as T (true).

This may sound crazy, but it greatly extends the power of Lisp. It means that any object in Lisp can not only contain data but can also have a true / false value associated with it. This turns out to be a very useful property when doing advanced programming, especially in creating databases or artificial intelligence systems.

When Logo was first designed, there was some discussion about whether or not to retain this Lisp feature. At the time, it was felt that it would be wiser to use only the literal words TRUE and FALSE to indicate Boolean conditions. It was hoped that this would give Logo greater clarity and simplicity without sacrificing anything important. Some recent versions of Logo (ExperLogo and TLC-Logo) have restored the Lispish view of "truth."

Each month I will answer a selected question about Logo programming techniques or the thinking which leads to programming. If you have a question, send it to: Jim McCauley, NLX Q and A, 876 East 12th Avenue, #4, Eugene, OR 97401. Please enclose a stamped self-addressed envelope. You may also send questions to me at 70014,1136 on CompuServe's electronic mail system.

Jim McCauley is a graduate student at the University of Oregon, studying with David Moursund, and has written Logo articles for many national publications.

Brian Harvey's Second Volume Announced

The second volumn in Brian Harvey's Logo series, *Projects, Styles, and Techniques*, from the MIT Press, will be available at the Logo 86 Conference.

NLXual Challenges

by Robs Muir

Objective Orientation

The chief problem with writing a monthly article such as this is the everpresent danger that the Fun (with a capital "F) of Logo will vanish in the glare of the DEADLINE - somewhat like the student who hasn't finished enough Logo task cards by the end of the term. I am happy to report that this month's challenge has me more excited about Logo's potential than I was when I first booted Terrapin Logo (version 1.0) on my Apple II!

Among the cognoscenti in computerdom, object-oriented programming is a current fascination. If you haven't read much about object-oriented programming, there is a very simple reason; there aren't many languages that support it! But this style of programming is emergent and may have strong implications for the next generation of languages.

What Is an Object?

Quite simply, an object is computational data structure that behaves like a computer within a computer. One object you are already familiar with is a turtle; you can send it messages, such as FD 50 RT 90, and the turtle-object will perform the task without you explicitly calculating where its new position will be. One "talks" to a turtle without regard to how it computes the result. A turtle uses its own *methods* for interpreting a *message*.

An object could just as well be a user-defined procedure that could return a value or respond to a *message* sent to it. Object-oriented programming allows for such an environment. One need not know how an object "does its thing;" each object represents a much higher degree of modularity than does a simple procedure. This modularity is vital to modern programming and may influence the forthcoming multi-processing computer architectures of the next decade.

The ideas for object-oriented programming were first supported by a language called Simula, developed by Alan Kay. Later, the concept was developed principally during the SmallTalk project at Xerox's Palo Alto Research Center (PARC) (see Goldberg, A., and Robson, D. *SmallTalk-80: The Language and Its Implementation*. Addison-Wesley, 1983). At the heart of SmallTalk is the concept of "message passing," where one object may transmit data to another only through the SENDing of messages.

Unfortunately, excursions into the world of objects are not for the faint of heart. Terms such as classes, methods, instances, and inheritance make learning about object-oriented programming something of a hurdle. However, at the Logo 85 conference, Mark Guzdial presented a set of LCSI Logo extensions, called OOLOGO (Object Oriented Logo), which provides an opportunity to experiment with SmallTalk's paradigm.

Read All about It

Recently, I was presented with a paper by Gary Drescher describing a fresh presentation of object-oriented programming. Gary was instrumental in the creation of ObjectLogo at the now defunct Atari Cambridge Research Center. ObjectLogo has all the power of traditional Logos with the addition of objects. Gary's paper, "Object-oriented Logo" (draft of August 25, 1985), describes the form of ObjectLogo, and presents a syntax that simplifies the concept of object-programming. This "function-linked" approach is very different from (and more natural than) the "message-passing" approach taken in SmallTalk-like programming.

Here is a simple example. In ObjectLogo we can create an object called "Bill.Bird which will politely speak with us. We will first need to create it, i.e., make it a something.

```
?MAKE "Bill.Bird :SOMETHING
```

Now we can tell it to have a name.

```
?TELL "Bill.Bird [HAVE "Name "Bill.Bird]
```

Next, we can tell "Bill.Bird to learn a procedure called Say. Note the resemblance to Logo's DEFINE.

```
?TELL "Bill.Bird [ HOWTO Say [ [ Msg ] [
PRINT (SE :Name "says :Msg) ] ] ]
```

If we now ask Bill.Bird to perform, we should get some response.

```
?TELL "Bill.Bird [ Say "Hi ]
Bill.Bird says Hi.
?TELL "Bill.Bird [ Say [ Hello There. ]
?Bill.Bird says Hello There.
```

Without much explanation, watch this:

```
?MAKE "Oscar.Object KINDOF "Bill.Bird
?TELL "Oscar.Object [ HOWTO Say [ [
MSG ] [ USUAL [ Say :msg ] ] [ PRINT [ Now I
say, SCRAM! ] ] ] ]
?TELL "Oscar.Object [ Say [ Hello
There. ] ]
Bill.Bird says Hello There.
Now I say, SCRAM!
```

Interesting, eh?

NLXual Challenges continued

ObjectLogo is about to be released for the rest of us who happen to own Macintoshes. Written by Coral Software Corporation, this new version of Logo promises to be the most powerful Logo yet released. But you don't have to wait to start using ObjectLogos just because you don't have a Mac.

I sent a copy of Gary's work to Mark Guzdial. Mark was so entranced with the draft of Drescher's article that he built a set of Logo extensions for users of Apple Logo //. Called Pseudo Object-Oriented Logo (POOL), it is not as fast or bug-free as a commercial implementation, but it is great fun as well as being a fantastic learning tool.

During the past week, I have taken Mark's program and modified it for Microsoft's Logo for the Macintosh, as well as for the original Apple Logo. These Logo procedures are free for the asking from the NLX.

Why would you want to get this set of procedures? Here's one reason why. Guess what this set of ObjectLogo commands will do:

```
?MAKE "TURTLE1 KINDOF "TURTLE
?MAKE "TURTLE2 KINDOF "TURTLE
?TELL "TURTLE1 [ REPEAT 4 [FD 50 RT 90] ]
?TELL "TURTLE2 [REPEAT 36 [FD 10 LT 10]]
```

Yes! Multiple turtles! I have had at least four turtles running on a 64K Apple using Apple Logo!

The Object of the Challenge

This month's challenge is simple. Mark Guzdial has agreed to make his source code available to interested Logophiles. Discover what you can do with Pseudo Object-Oriented Logo and share your experiences with us. Send a *stamped* (39¢) self-addressed envelope to:

NLXual Challenges - POOL
Attn: Robs Muir
1688 Denver Ave.
Claremont, CA 91711

We will send a program listing for Apple Logo, Apple Logo //, and MacLogo, along with some preliminary tips for using Pseudo Object-Oriented Logo.

In addition, Coral Software Corporation has agreed to provide final drafts of Gary Drescher's paper, "Object-Oriented Logo." In the paper, Gary also proposes how ObjectLogos might be used with children, including a suggested sequence of introduction. For a copy of Gary's paper, send a *stamped* (39¢) self-addressed envelope to:

Coral Software Corporation - Dresher paper
Attn: Jeremy Jones
PO Box 307
Cambridge, MA 02142

(While you're at it, you might also request some literature on their soon-to-be released ObjectLogo.)

Please send your experiences with this program so that we can continue to improve its performance. I can also be reached via CompuServe [70357,3403] or BITnet [MUIRR@CLARGRAD].

Try herding turtles, just for fun!

Robs Muir is a teacher in the Claremont (CA) Unified School District and Claremont Graduate School.

Logo Notes

MusicLogo is now available from Terrapin, Inc. This set of Logo language and programs was developed in conjunction with Jeanne Bamberger, a noted Logo music advocate and music theorist from the Massachusetts Institute of Technology. The product combines the power of Logo with the potential of the ALF synthesizer, resulting in control over six voices, each with a range of five octaves. The package includes a MusicLogo language disk and backup, a projects disk, and a 190-page sourcebook. The ALF synthesizer must be purchased separately. The package costs $99.95. Write to Terrapin, Inc., 222 Third Street, Cambridge, MA 02142, or call (617) 492-8816.

Also available from Terrapin is *The Logo Project Book: Exploring Words and Lists*, by Alison Birch. An activities disk is included. The cost is $19.95.

The National Logo Exchange, copyright © 1986 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, $25 (US) per year mailed first class from Charlottesville, VA. $5 (US) additional per year for addresses outside the United States, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of $2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01971. Address editorial and subscription related correspondence to: The National Logo Exchange, PO Box 5341, Charlottesville, VA 22905. Published since September 1982. ISSN 0734-1717. CompuServe 70020,223.

Editor..Tom Lough